A computational model of short-term recognition and recall

Thesis

presented to the Faculty of Arts and Social Sciences

of the University of Zurich

for the Degree of Doctor of Philosophy

by **Hsuan-Yu Lin**

Accepted in the spring semester 2016

on the recommendation of the doctoral committee:

Prof. Dr. Klaus Oberauer (main advisor)

Prof. Dr. Björn Rasch

Zurich, Switzerland, 2016

Abstract

In this study, a connectionist model for serial recall– Serial-Order in the Box (SOB)(Lewandowsky & Farrell, 2008; Oberauer, Lewandowsky, Farrell, Jarrold, & Greaves, 2012) – is extended for explaining data from short-term recognition. The main motivation behind the extension is to create a computational model which can simulate both short-term recall tasks and short-term recognition task, which in turns bridging both tasks under the same memory representation. In SOB, the memory representation consists of bindings between items and contexts (i.e. position markers of serial positions). The structure of the network and encoding process are kept unchanged for the recognition model, while the retrieval process for the recognition task differs from the serial-recall task. The retrieval process is modeled as comparing the probe to the memory content retrieved from the context, and the context used for retrieving the memory content is retrieved by activating the probe and deblurred through the recognition process. At the beginning of the retrieval process, the context used for retrieval is noisy because of the superposition between bindings, and it is subsequently sharpen to the context which is most strongly associated to the probe. Thus, the retrieved memory content is a mixture of all the memory items at the beginning and then is gradually narrowed down to the memory content which is most similar to the probe. The recognition model is able to simulate the set-size effect, the serial-position effect, and the speed-accuracy trade-off in both Sternberg's memory scanning task. The model is also able to simulate the expectation effect for the incoming task, the local recognition task, and the performance from continuous stimulus recognition task. This is also the first computational model explaining both recognition and serial recall of information in working memory.

Zusammenfassung

In dieser Studie wird ein konnektionistisches Modell für seriellen Abruf - das Serial-Order in the Box Modell (SOB: Lewandowsky & Farrell, 2008; Oberauer, Lewandowsky, Farrell, Jarrold, & Greaves, 2012) - erweitert, um Daten aus der kurzfristigen Wiedererkennung zu erklären. Die Hauptidee dabei besteht darin ein computationales Modell zu erstellen, welches beide Aufgaben simulieren kann, den kurzfristigen Abruf (Recall) und die kurzfristige Wiedererkennung (Recognition), was wiederum die beiden Aufgaben in derselben Gedächtnisrepräsentation vereinen würde. In SOB besteht eine Gedächtnisrepräsentation aus Verbindungen zwischen Inhalten und Kontexten (z.B. Positionsmarker für die Position im seriellen Ablauf). Die Struktur des Netzwerkes und der Endcodierungsprozess werden für das Wiedererkennungsmodell nicht verändert, jedoch ist der Abrufprozess für die Recognition-Aufgabe anders als für die serielle Recall-Aufgabe. Der Abrufprozess wird modelliert als das Vergleichen einer Probe mit dem Gedächtnisinhalt der aus dem Kontext gewonnen wird. Dieser Kontext wiederum wird durch das Aktivieren der Probe gewonnen und wird geschärft durch den Wiedererkennungsprozess. Zu Beginn des Abrufprozesses ist der Kontext, welcher für den Abruf genutzt wird, noch unscharf, weil die Verbindungen überlagert sind, er wird jedoch geschärft durch den Kontext, der am stärksten mit der Probe assoziiert wird. Folglich ist der abgerufene Gedächtnisinhalt eine Mischung aus anfänglich allen Gedächtnisinhalten und wird später geschärft zu dem Gedächtnisinhalt, der der Probe am ähnlichsten ist. Dieses Wiedererkennungsmodell ist in der Lage den Set-Size-Effekt, den Serial-Position-Effekt, sowie den Speed-Accuracy-Trade-Off in beiden Sternberg’s Memory Scanning Aufgaben zu simulieren. Das Modell ist auch in der Lage den Erwartungseffekt auf die kommende Aufgabe, die Local-Recognition-Aufgabe und die Leistung in einer Wiedererkennungsaufgabe mit einem ununterbrochenen Stimulus zu

simulieren. Zusätzlich ist dieses Modell das erste Modell, welches sowohl Wiedererkennung als auch seriellen Abruf von Information im Arbeitsgedächtnis eklären kann.

Table of Contents

1 Introduction 9

1.1 The existing recognition models 10

1.1.1 Phenomena simulated by the existing recognition models 10

1.1.2 Noisy Exemplar Model 15

1.1.3 Exemplar-based Random-walk Model 17

1.1.4 Iterative-resonance Model 19

1.2 The missing mechanisms in previous models 22

1.2.1 Activation strength across serial positions 22

1.2.2 The decision threshold in NEMO and EBRW 23

1.3 Adapting Serial-Recall Models 24

1.4 SOB-CS 26

2 SOB-R 29

2.1 Verbal Description of SOB-R 29

2.1.1 Shadow of previously used context 30

2.1.2 Assumptions about the recognition process 32

2.2 Content and context representation 34

2.3 Encoding process 37

2.4 Serial-recall process 39

2.4 1 Simulation of serial-recall task 41

2.5 Recognition process 42

2.5.1 Simulation of Sternberg's task 46

3 Simulating other tasks 51

3.1 Speed-accuracy trade-off 51

3.2 Local recognition task 53

3.3 Extralist-feature effect 61

3.4 The expectation effect on the serial-position curve 63

3.4 Continuous stimulus 67

4 General Discussion 71

4.1 Failed simulations and ways to fix them 71

4.1.1 Linear increasing of the reaction time 72

4.1.2 Set-size effect on intrusion cost 73

4.1.3 Extralist-feature effect 76

4.1.4 The expectation effect on the serial-position curve 77

4.2 Comparing SOB-R to the other recognition models 80

4.2.1 Activation strength between serial positions 81

4.2.2 Constant threshold between set sizes 81

4.2.3 Simulating the local recognition task 82

4.3 Conclusion and Outlook 83

5 References 86

Introduction

Since Sternberg (Sternberg, 1969) published one of his famous studies, cognitive psychologists conducted research on short-term recognition for more than half a century. Besides the linear increase of the set-size effect on reaction time found by Sternberg, many other phenomena about recognition were found. However, despite having a rich amount of data, the process behind the recognition task remains unclear and is still under debate (Cowan, Rouder, Blume, & Saults, 2012; Gilchrist & Cowan, 2014; Wixted, 2007; Yonelinas, 2002). Many theories with fundamentally different assumptions are able to explain those phenomena. Often, implementing those theories into computational models would force those theories to explain recognition process in greater detail and provide more detailed predictions. However, there are not many computational models of short-term recognition. The existing models focus on some aspects of the recognition process but leave some important details unexplained. Therefore, in this dissertation, my goal is to create a computational model for short-term recognition, which aims for not only simulating the phenomena in short-term recognition but also filling in the details which were left unexplained by other computational models. Also, the model serves as a bridge between two similar tasks with different kind of response: short-term serial-recall task and short-term recognition task.

In this dissertation, I will begin with reviewing the current existing computational models including their unique features and missing mechanisms. I will propose my solution for those missing mechanisms by borrowing the mechanisms from a serial-recall model, Serial-Order in a Box (SOB-CS, Oberauer, Lewandowsky, Farrell, Jarrold, & Greaves, 2012). Then I will present my short-term recognition model, Serial-Order in a Box-Recognition (SOB-R), as an extension of SOB-CS, followed by simulations of the model for the phenomena in short-term recognition. Finally, I'll discuss SOB-R and compare it with other recognition models.

1.1 The existing recognition models

Although short-term recognition has been studied for more than half a century, and there are many theories regarding the possible process behind recognition, there are not many computational models. Some models were not proposed as a complete recognition model but are designed for testing specific assumptions in theories and focus on certain diagnostic phenomenon for specific task, e.g. the models proposed by Oberauer (2008). Some models explain the results from recognition tasks but their main emphasis is on the memory storage while neglecting the recognition process, e.g. the slot model of working memory capacity (Awh, Barton, & Vogel, 2007). Some models incorporate the representation of the stimulus, the memory storage, and the recognition process together and explain a broader range of tasks. Because I aimed for SOB-R to achieve this level of detail while also being able to simulate the phenomena previous models can simulate, I will discuss those general recognition models. To my knowledge, the existing general short-term recognition models are: The Noisy Exemplar Model (NEMO; Kahana & Sekuler, 2002), the Exemplar-based Random-walk Model (EBRW; Nosofsky, Little, Donkin, & Fific, 2011), and the Iterative-resonance Model (IRM; Mewhort & Johns, 2005). Those recognition models do not only simulate the Sternberg's task, many other phenomena are also simulated by the models to support the mechanism behind the models. Thus I'll first introduce those phenomena simulated by NEMO, EBRW, and IRM, followed by introducing the models individually.

1.1.1 Phenomena simulated by the existing recognition models

Before introducing the short-term recognition models, I will first introduce the phenomena simulated by the recognition models. All the models are capable to simulate the set-size effect and the serial-position effect of the Sternberg task. EBRW is able to simulate the speed-accuracy trade-off function of the recognition task. IRM can simulate the extralist-feature effect.

All the effects simulated by the models are observed from the standard Sternberg task. The Sternberg's short-term memory scanning task is one of the most studied paradigm in the short-term recognition. The procedure of the Sternberg task is following: At the encoding phase, the memory items are presented sequentially. After all the memory items were presented, an end-of-list signal is presented to inform participant that the encoding phase is finished. A probe is then presented on the screen. Participants are asked to judge whether the probe is in the memory items or not as quickly and accurately as possible. The probe can either be a positive probe, a probe which is in the memory list, or a new probe, a probe which is not in the memory items. Participants are supposed to accept the positive probes and reject the new probes. The dependent variables of the Sternberg task are the reaction time and the proportion of correct responses. In most cases, the proportion of correct responses is at ceiling, thus the reaction time is often the index of participants' performance.

Set-size effect in the Sternberg task

The typical finding of reaction time in the Sternberg task is the set-size effect. The reaction time for correctly accepting the positive probes and rejecting the negative probes increases linearly with set sizes. The amount of increasing is the same in both positive probes and negative probes, as shown in Figure 4.

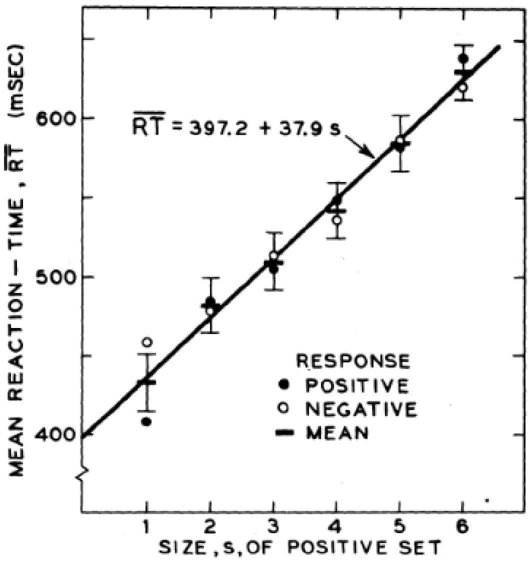

Figure 1. The typical set-size effect from Sternberg's task. The set-size effect is the Figure 4 of Sternberg (1969).

Serial-position effect in Sternberg's task

A serial-position effect is also commonly observed in the Sternberg task. The serial position of the positive probe affects the reaction time of the probe. The typical serial-position effect is shown in Figure 2. The performance is better at the beginning and the end of serial positions comparing to the performance at the middle of the serial positions. The serial-position effect can be described as the primacy gradient and the recency gradient. The primacy gradient is the decrease of performance from the beginning of the serial position to the end of serial position. The recency gradient is the decrease of performance from the end of serial position to the beginning. In Sternberg's task, the recency gradient is often stronger than the primacy gradient.

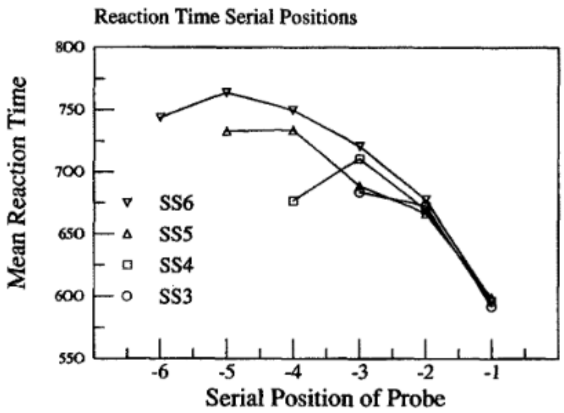

Figure 2. The typical serial-position effect observed in the Sternberg's task. The serial-position effect is from the Figure 9 of McElree & Dosher (1989).

Speed-accuracy trade-off function

A speed-accuracy trade-off function is observed when participants are under time pressure. When participants are under time pressure, they tend to sacrifice the accuracy of their responses in order to successfully response under the time pressure. The speed-accuracy trade-off function is often regarded as reflecting the change of quality of evidence during the response process. Speed-accuracy trade-off function is measured with the response-deadline paradigm.

In the response-deadline paradigm, participants are instructed to make a response immediately whenever they heard an audio cue. Typical findings from the response-deadline paradigm show that participants' accuracy increases when the interval between stimulus and response deadline increases. With the shortest interval, i.e. the audio cue appears immediately after the stimulus; participants' accuracy level is at chance. With the interval getting longer, participants' accuracy increases. The increase decelerates and eventually reaches an asymptote. The rate of increasing and the asymptote are often treated as the important parameters in the speed-accuracy trade-off function.

Previous studies have shown that the typical pattern of speed-accuracy trade-off curve can also be observed in Sternberg's task with response-deadline paradigm (McElree &

Dosher, 1989). Moreover, different set sizes show different speed-accuracy trade-offs. Smaller set sizes have higher asymptotes and larger rates compared to larger set sizes, as shown in the left side of Figure 3. The speed-accuracy trade-off function also varies at different serial positions. The most recent item has the sharpest increasing rate and highest asymptote in the speed-accuracy trade-off function. The rate and the asymptote decrease when the probe originates from earlier in the list, as shown in the right side of Figure 3. The speed-accuracy trade off function can be simulated by EBRW.

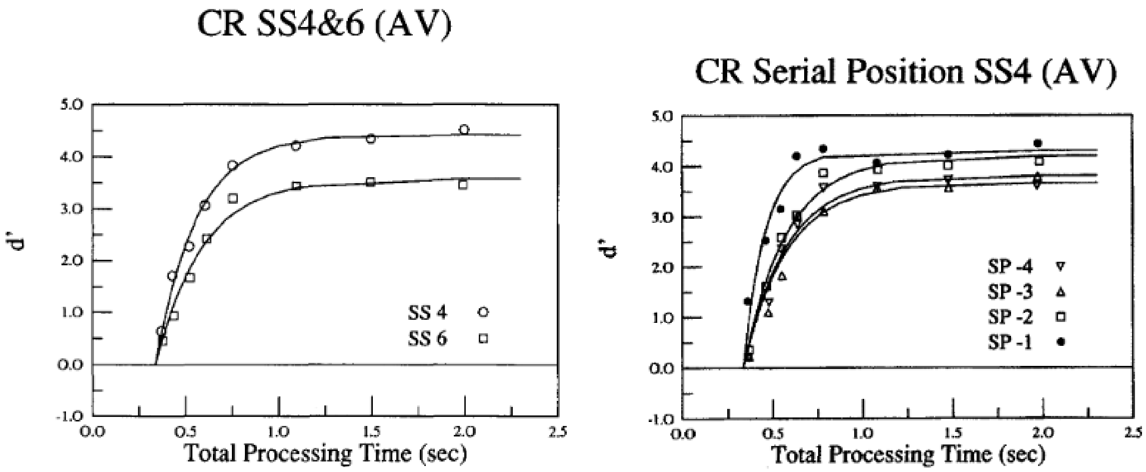

Figure 3. The speed-accuracy trade-off function for different set sizes (left side) and the speed-accuracy trade-off function for different serial positions (right side). The left figure is from the Figure 13 of McElree & Dosher (1989). The right figure is from the Figure 12 of McElree & Dosher (1989).

Extralist-feature effect

The extralist-feature effect is the finding that the ability of rejecting a new probe is determined by the number of extra-list features of that probe (Mewhort & Johns, 2000). In an experiment by Mewhort & Johns (2000), the similarity between probe and the studied items was manipulated. Each item comprised of two features. The memoranda consisted of four items: *Aa*, *Ab*, *Bc*, *Cc*. The upper case letter indicates the content of the first feature, and the lower case letter indicates the content of the second feature. Thus, *Aa* and *Ab* share the same *A* content for the first feature but have different content for the second feature. Four types of new probes were tested with different similarity to the memoranda. The first type of new probe is the 0:0 probe, e.g. *Xx*, where *X* and *x* represent extra-list features, the content of these features was never presented in the memoranda. The second type of new probes is the 1:0 probe, e.g. *Xa, Xb, Bx,* and *Cx.* One feature in the probe was presented once in the memoranda, and another feature was an extra-list feature. The third type of new probes is the 2:0 probe, e.g. *Ax* and *Xc*. One feature was presented twice in the memoranda, and another feature was an extra-list feature. The forth type of new probes is the 1:1 probe, e.g. *Ba, Ca, Bb,* and *Cb*. Two features were presented once in the memoranda. The study showed that 0:0 probed were the easiest and 1:1 probes were the most difficult probes to be rejected. The performance of 1:0 probe and 2:0 probe were at a similar level, as shown in Figure 4. The extralist-feature effect can be simulated by IRM.

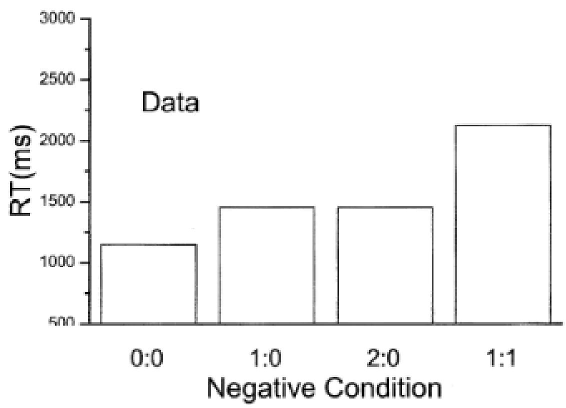

Figure 4. The observed extralist-feature effect. The figure is from Figure 1 of Mewhort & Johns (2003).

1.1.2 Noisy Exemplar Model

The Noisy Exemplar Model (NEMO, Kahana & Sekuler, 2002) is a recognition model based on the Generalized Context Model (Nosofsky & Palmeri, 1997). As GCM, NEMO assumes items (exemplars) exist in a multidimensional space, and the similarity between items is represented as the distance between items in the multidimensional space. Unlike other recognition models, which only consider summed similarity between probe and items for the old/new judgment in the recognition process, NEMO also takes the average inter-item similarity into account. When the average inter-item similarity increases, larger similarity between probe and the items is required for accepting the probe. Through taking the inter-item similarity into account in the recognition process, NEMO is able to explain the set-size effect and the serial-position effect in the recognition task.

Recognition process in NEMO. Like GCM, NEMO assumes each item exits in a multidimensional space where each dimension represents a feature of the items. Each feature dimension can be expanded or compressed depending on the importance of the feature, which varies from task to task. The location of an item in the multidimensional space is determined by the value of each feature. The distance between two items determines the similarity between these two items. The similarity between two items decreases when the distance between items increases, since either one or multiple features of items are more dissimilar. The outcome of the recognition process in NEMO is determined by three factors: 1) The similarity between probe and items, 2) the average inter-item similarity, and 3) the activation strength of each item.

NEMO assumes that the probe also exists in the multidimensional space. The summed distance between the probe and each item determines the summed similarity of the probe. The probe is more dissimilar if the summed distance is larger. For positive probes, since the probe is exactly the same as one of the items, the summed distance is reduced because one of the distances between the probe and the items is zero, which results in higher similarity. On the other hand, the location of a new probe in the multidimensional space differs from the locations of all the items. The summed distance of a new probe is larger than the summed distance of a positive probe, which results in less similarity. Thus, by using the summed similarity between probe and items, NEMO is able to distinguish the difference between positive probes and new probes. The summed similarity is compared to a decision threshold for accepting or rejecting the probe. If the summed similarity of probe is larger than the threshold, the probe will be accepted. If the summed similarity is smaller than the threshold, the probe will be rejected. The decision threshold varies at different set sizes. NEMO simplified the placement of the decision thresholds and assumed the optimal thresholds for separating positive probes and new probes at different set sizes.

Besides the summed similarity between probe and items, NEMO also takes average inter-item similarity into account. The average inter-item similarity is the average pairwise similarity between items. If the memory items are more similar to each other, the average inter-item similarity is larger, and a result the recognition judgment requires more similarity to accept probes. The summed similarity of probe and the average inter-item similarity are both influenced by the activation strength of items. The activation strength of items varies depending on the serial position of the item. The more recent items have higher activations compared to the items from the beginning of the list. The items with higher activation have higher weight when determining the similarity, thus the more recent items have higher weight than distant items, which results in the recency effect. Moreover, the positive probes in larger set sizes are more likely to originate from the early positions of the list, thus on average, the positive probes have lower activations in the larger set size, which results in the set-size effect of positive probe.

1.1.3 Exemplar-based Random-walk Model

The Exemplar-based random walk (EBRW) model was originally designed for categorization tasks (Nosofsky & Palmeri, 1997) and later was adopted to recognition tasks (Nosofsky, Cox, Cao, & Shiffrin, 2014; Nosofsky et al., 2011). EBRW assumes that each item (exemplar) takes a place in a multidimensional space. The distance in the multidimensional space represents similarity, which decreases when distance increases. While recognizing a probe, the sum of similarity between the probe and the exemplars, i.e. the sum of distance between the probe and the exemplars in the multidimensional space, is used for old/new judgment. EBRW is able to simulate most of the phenomena in short-term recognition task. Combined with the random walk mechanism, EBRW is also able to simulate the distribution of response times and the speed-accuracy trade-off.

Recognition process in EBRW. Similar to NEMO, EBRW assumes each exemplar exists in a multidimensional space where each dimension represents a feature of the exemplar. The location of an exemplar in the multidimensional space is determined by the features of the exemplar. The similarity between two exemplars decreases when two exemplars are further away from each other, since the features of the exemplars are more dissimilar. During recognition, each exemplar contributes a certain amount of activation to the probe. The amount of activation from an exemplar is determined by the similarity between the probe and the exemplar, and also the memory strength of the exemplar. With more similarity between the probe and the exemplar, a larger amount of activation is contributed. Higher memory strength of the exemplar also contributes larger activation. The sum of activations from exemplars represents the activation of the probe. The activation of positive probes is larger than the activation of negative probes, since a positive probe is identical to one of the exemplars. Thus the model is able to tell the difference between positive probes and negative probes based on the difference in activation.

The reaction time and the probability of response is determined by a random-walk process with the activation of the probe. The random-walk process is similar to the Diffusion model (Ratcliff, 1978). An accumulator initiates at the center between two response boundaries: +OLD and –NEW, which represent the response of accepting the probe as an old item and rejecting the probe, respectively, and randomly walks toward one of the two boundaries at each time frame. The probability of moving toward +OLD or -NEW is determined by the activation of the probe and the baseline activation. The activation of probe provides the evidence for the accumulator to move toward the +OLD boundary, and a baseline activation provides the evidence for the accumulator to move toward the -NEW boundary. Once the counter reaches one of the two boundaries, a response is made based on the boundary reached. The reaction time is simulated as the number of time steps required to reach one boundary. The higher the probe activation is, the more likely the counter is to walk toward +OLD, thus the response is more likely to be +OLD, and occurs quicker.

Because the baseline activation in the random-walk process increases with set sizes, EBRW is able to simulate the set-size effect. The positive probe has to overcome the increase of baseline activation over set sizes, which results in increasing reaction time for accepting the positive probe. The activation of new probe increases with set sizes, because every item in the multidimensional space contributes a small amount of activation to new probe. Although the baseline activation increases with set sizes, because the activation of new probes increases at a larger rate comparing to the baseline activation, the reaction time of successfully rejecting the new probe increases accordingly. EBRW also assumes the memory strength differs at different serial positions. The memory strength of an item is determined by the primacy gradient manipulator, i.e. the memory strength gradually decreases toward the later serial positions, and the recency gradient manipulator, i.e. the memory strength gradually increases toward the end of the serial positions. Because the memory strength varies at different serial positions; the activations of positive probes differs according to the serial positions. Thus the model also simulates the serial-position effect.

1.1.4 Iterative-resonance Model

The Iterative-resonance model (IRM, Mewhort & Johns, 2005) is a recognition model based on MINERVA2 (Hintzman, 1984). The recognition process is assumed to consist of the comparison between the probe and an echo, which is the retrieved memory. In addition, IRM assumes the recognition process begins with global comparison and gradually changes to local comparison. The probe is accepted if the echo is similar enough to the probe, or rejected if there are enough mismatching features. IRM is able to simulate the set-size effect and the serial-position effect in Sternberg's task. Also, IRM can simulate the extralist-feature effect.

Recognition process in IRM. In IRM, items are represented in a distributed fashion. An item is represented by a vector, and the elements in the vector represent a feature or a group of features of the item. The similarity between items and the mismatch between items are measured differently. The similarity between two items is measured though the summed product between individual features, i.e. the dot-product. The mismatch is measured by counting the number of mismatching elements for which the difference between the elements in the probe and the elements in the echo passes a certain threshold. The difference between similarity and mismatch is that similarity takes the difference between all elements into account while the mismatch only counts the number of mismatching elements which surpass the threshold. Assume two items match with regard to all their features except for one mismatching pair of elements. As long as the difference between the element of probe and the element of echo is larger than the threshold, it counts as one mismatch regardless of the magnitude of the difference. However, because the similarity is the summed product, the magnitude of the difference matters. If the magnitude of the difference of the mismatching elements is large, even when all the other elements are perfectly matched, the similarity between items would be relatively low.

The recognition process in IRM is based on the comparison between the probe and the retrieved memory, which is called echo. The old and new responses are made on the bases of different thresholds. The probe is accepted as an old item when the similarity between probe and echo reaches the acceptance threshold. The probe is rejected as a new probe when the amount of mismatch between echo and probe reaches the rejection threshold. In addition to assuming different response thresholds for accepting and rejecting probes, IRM also assumes that the echo changes over time during the recognition process. At the beginning of the recognition process, the echo is the average of all the items in the memory, which effectively makes the comparison between echo and probe a global comparison. As time goes by, the echo gradually sharpens according to the degree of match between the probe and the memoranda. The sharpening process is driven by the resonance between the probe and items in the memory, such that the echo is updated based on the similarity between the probe and the items in the memoranda. The more similar item contributes with a stronger weight to the echo. If the probe is a new probe, the resonance between probe and items would be minimized, thus the echo would end up as nothing, which causes the number of mismatches to increase, leading to rejection of the probe. If the probe is an old probe, the probe will resonate with the matching item, which sharpens the echo toward the matching item and effectively turns the comparison between echo and probe from global match into local match. At the beginning of the recognition process, the echo comprises the sum of all the items in the memoranda, thus the comparison between the echo and the probe is the comparison between the probe and all the items, which is the global match comparison. After the echo is sharpened to the most matching item, the comparison between the echo and the probe becomes the comparison between the most matching item and the probe. The rest of the items in the memory are not involved in the comparison between the echo and the probe, thus the comparison is a local match comparison. The similarity between echo and positive probe increases when the sharpening process happens, which results in accepting the probe.

IRM accounts for the set-size effect and serial-position effect by assuming the activation of items varies across the serial positions. The more recent items have stronger activation comparing to the items in the earlier of the list. For items with lower activation, the sharpening process takes longer to sharpen into the matching item, which results in longer reaction times. Because the rejection of new probes is the result of the number of mismatching features, IRM can also account for the extralist-feature effect.

1.2 The missing mechanisms in previous models

NEMO, EBRW, and IRM are able to explain many fundamental phenomena in short-term recognition. However, those models leave some mechanisms unexplained. Though those mechanisms might not be the main focus of the models, missing mechanisms still play important roles in the phenomena the models can simulate. All the models fail to explain the cause of different activation for items across serial positions, which results in the serial-position effect. NEMO and EBRW also rely on a decision criterion which changes linearly with set sizes, which produces the set-size effect. Those missing mechanisms are crucial for the models to function but were left unexplained.

1.2.1 Activation strength across serial positions

NEMO, EBRW, and IRM assume that items have different activation values across the serial positions. However, the activation in all the models is assumed as the result of primacy gradient and recency gradient without specifying the mechanisms behind primacy and recency gradients. In NEMO and IRM, the recency gradient is described by a power function, and for the primacy gradient it is assumed that first item receives additional strength. In EBRW, the activations are assumed as free parameters across serial position. One later study found that a power function is best for describing the activation across serial position (Donkin & Nosofsky, 2012). Although all the models require different activations across serial positions to simulate the serial-position effect, the models do not explain the mechanisms behind the different activation values. Though EBRW was not designed to explain serial-position effect but focused on the multi-dimensional feature space and item representation, because the serial-position effect is a robust and important finding in short-term recognition, EBRW left an important benchmark unexplained. The same applies to NEMO and IRM which left serial-position effects unexplained.

1.2.2 The decision threshold in NEMO and EBRW

NEMO and EBRW cannot simulate the set-size effect without letting the criterion vary. NEMO and EBRW rely on summed similarity as evidence of recognition: The activation of the probe is the sum of the similarity between probe and memoranda. The need of multiple decision thresholds in summed-similarity models can be easily explained by a simplified summed-similarity model. Assuming the similarity between probe and the perfect matching item is s, and the average similarity between probe and mismatching item is b. The model uses summed similarity for probe judgment. Thus, the summed similarity for positive probe and negative probe at set size n is $s + b*(n-1)$ and $b*n$, respectively. The summed similarity is then compared to a fixed decision threshold t, and the performance is determined by the difference between the summed similarity and the decision threshold. Ideally, the decision threshold t should be bigger than the similarity of negative probes and smaller than the similarity of positive probes. Therefore, the performance of positive probe in set size n is a function of $s + b*(n-1) – t$, and the performance of negative probe in set size n is a function of $t – b*n$. Depending on the task, the averaging similarity between probe and mismatching item has three possible cases of values. The first case is $b > 0$. This means that even though the probe does not match the item, the probe and the item are similar enough for affecting the response toward "yes". Thus for positive probes, the performance $s + b*(n-1) – t$ is larger when list length n increases. For negative probes, the performance $t – b*n$ is smaller when n increases. The second case is $b < 0$; which means that the probe and mismatching items are very dissimilar, which affects the response toward "no". Thus, the performance of positive probe $s + b*(n-1) – t$ is smaller when n increases. The performance of negative probe $t – b*n$ is larger when n increases. In both cases, the list length has opposite effects on the performance of positive probes and negative probes, inconsistent with previous findings (Sternberg, 1969). The last case is $b = 0$. This means that the mismatch between probe and item does not affect the response at all. The response is solely determined by whether there is a match or not. The performance of positive probe can be simplified as *s* - *t*, and the performance of negative probe is *t*. Because the performance of both positive probe and negative probe are not affected by list length *n*, there is no set size effect in this case. In either case, a summed-similarity model could not simulate the set-size effect if the decision threshold is constant.

The decision threshold in NEMO and EBRW is assumed to vary at different set sizes. In NEMO, the decision thresholds are set to be the optimal thresholds for separating the positive probes and the new probes at different set sizes. Depending on the distribution of positive probes and the negative probes, an optimal decision threshold can simulate many possible set-size effects, which gives NEMO extreme flexibility. In EBRW, the baseline activation which drives the response toward rejection increases linearly across set sizes, which is one of the factors to simulate the linear increasing of set size effect.

1.3 Adapting Serial-Recall Models

Though recall and recognition memory are normally addressed separately, these two kinds of memory are unlikely to be completely separated. In the field of short-term memory studies, the serial-recall task and the recognition task are rarely discussed together, but these two tasks share similar experimental procedures and findings. In serial recall, participants learn a series of items. Shortly after learned, participants are asked to recall the items they learned in the order they had learned them. In recognition, participants learn a series of items as in a recall task. However, participants do not have to recall all the items they learned. Instead, a probe is presented on the screen, and participants are asked to decide if the probe is in the series they learned. Besides the fact that both tasks have similar experiment procedures, previous studies show that the performance levels in serial-recall task and recognition task both are affected by the list length, and by the position of the tested item in the series of items. Moreover, the serial-position effect of recall task is similar to the serial-position effect of recognition task, after balancing the output order (Oberauer, 2003). According to the similarity of experimental procedures and findings in serial-recall and recognition, models for serial-recall tasks should be able to explain, to some degree, the findings in recognition tasks, and the other way around.

In this dissertation, I extend an existing serial-recall model to short-term recognition tasks and assume that the serial recall and recognition tasks share the same encoding process and underlying memory representation. Only the recognition process differs from the serial-recall model. The reason for extending an existing serial-recall model for recognition task instead of creating a recognition model is that there is, to my knowledge, no model which can simulate both serial-recall task and short-term recognition task, despite the similarity between tasks. Some research modified an existing serial-recall model for recognition task but did not develop it into a general recognition model (Hay, Smyth, Hitch, & Horton, 2007). By extending the existing serial-recall model, the new model can serve as a bridge between the two tasks.

Another advantage of extending an existing serial-recall model is that most serial-recall models explain both the encoding process and the process of serial recall. By inheriting the encoding process, the activation values across serial position can be explained, and it can be avoided to simply assume the primacy and recency gradient of activation values. For example, the Serial-order in the Box (SOB) model assumes that the primacy gradient across serial positions arises because of novelty-gated encoding: The more novel each item is, the stronger the item will be encoded. The Scale-Invariant Memory, Perception, and Learning model (SIMPLE) assumes that the distinctiveness between items is reduced when time passes, which results in a recency gradient across serial positions. By drawing on the serial-recall model with its explanation of the serial-position effect, the extended recognition model provides the missing mechanisms of NEMO, EBRW, and IRM.

1.4 SOB-CS

In this dissertation, I choose SOB-CS as the base model for extension. SOB-CS (Oberauer et al., 2012) is a well-developed neural-network model which assumes that the memory is represented as bindings between content (i.e. to-be-remembered item) and context (i.e. serial position). The encoding process is assumed as forming the bindings between content and context, and the recall process is assumed as activating each context to retrieve the bounded content. In SOB-CS, incorrect responses are explained by interference. There are two sources of interference: interference in context layer and interference in content layer. The interference in context layer arises because of the overlap between contexts. When recalling the content, a context is used as cue to retrieve the content bound to it. However, because the context is not represented precisely, activating a context also activates the other contexts which share similar features. The retrieved memory is not just the content bound to the cued context but also other contents bound to the non-cued contexts, which reduces the precision of recall. Content representations also overlap with each other, which causes the interference in the content layer. Overlapping in contents creates difficulty to distinguish between contents, which also reduces the precision of recall. With interference, SOB-CS is able to simulate many phenomena in serial-recall tasks, including the set-size effect, the serial-position effect, and the transposition gradient (which will be explained below), which are shared between serial-recall task and recognition task.

One major advantage to choose SOB-CS among other serial-recall models is that SOB-CS has one important mechanism: energy. Conceptually, energy represents the entropy in information theory (Shannon, 2001). In SOB-CS, energy represents the difference between the expectation about the upcoming item and the actual to-be-encoded item, and the expectation is the content retrieved by activating context, i.e.: While encoding, the expectation of to-be-encoded item is retrieved from memory via activating the serial position of the to-be-remembered item. The energy modulates the strength of binding and unbinding information. When the expectation does not match the incoming item, the strength of binding is higher, because the memory system has to compensate the mismatch, compared to the strength of binding when the expectation matches the outcome. SOB-CS utilizes energy-modulated strength to simulate the primacy gradient of encoding strength. While encoding the first item, since the only information stored in the memory is the random noise, the retrieved expectation consists of the random noise, which is dissimilar to the to-be-encoded items. Thus, the energy for the first item is the lowest, and the encoding strength of the first item is the strongest. After the model has encoded the first item, the retrieved expectation from the second serial position consists of the random noise and the first encoded item. The first item and the second item have some degree of similarity, thus the energy for the second item is larger than the energy of the first item, so that the encoding strength of the second item is smaller than the encoding strength of the first item. As more items are encoded in the memory, the retrieved expectation consists of more items encoded previously, and the encoding strength decreases as the serial position increases.

I assumed energy is also used for assessing the novelty of a probe during the recognition process. Since the energy is the result of comparing retrieved information from memory and an actual item, the recognition process can be viewed as comparison between the retrieved memory from memory and the probe, which is the same as measuring the energy of the probe. Thus, by extending the SOB-CS model to recognition, the energy in SOB-CS can also be used in recognition process.

The extended recognition model does not only aim for simulating a wide range of phenomena in the recognition tasks, but also as a bridge between serial-recall task and recognition task. The model has to be able to simulate both tasks and explain the difference between tasks at the same time. Since SOB-CS is able to explain most of the phenomena in the serial-recall task, the extension only has to focus on the recognition task while keeping the simulation of the serial-recall task unchanged. The extension should also keep the modifications of SOB-CS as little as possible to preserve the simulations of serial-recall tasks.

SOB-R

Serial-Order in the Box-Recognition (SOB-R) is a recognition model extended from SOB-CS with minimal modifications. Like its predecessor, SOB-R is a neural-network model which assumes memory in serial-recall and recognition task is represented as bindings between content and context. The encoding process is also unchanged, content and context are bound together with hetero-association. For the recognition process, I assume that the context is used to retrieve the content from the memory, and the retrieved memory is compared to the probe. The context used to retrieve the content, however, changes over time. At the beginning of the recognition process, the average of all the used contexts in the current trial is used to retrieve the content, which results in that retrieved memory is the average of all the memory items. Thus the comparison between retrieved memory and probe becomes global comparison. The activated context is then gradually sharpened into the context which gives the most similarity between the retrieved memory and the probe, which turns the comparison into local comparison.

In this chapter, I will introduce SOB-R and begin with the verbal description of SOB-R and followed by the formal description of SOB-R. I introduce the representation and encoding process in SOB-R. Since SOB-R includes a few minor modifications from SOB-CS, I will then introduce the serial-recall process of SOB-R and simulate the serial-recall task with the modifications to ensure the simulation with serial-recall task still holds. Last, I will introduce the recognition process in SOB-R and the simulation of the most basic recognition paradigm: Sternberg's task.

1.1 Verbal Description of SOB-R

SOB-R inherits most of the assumptions in SOB-CS. SOB-CS assumes content, i.e. memory item, and context, i.e. serial position, are represented in a distributed fashion. The encoding process is assumed as forming bindings between content and context, e.g. remembering certain item is presented at certain serial positions. The strength of binding is modulated by energy, i.e. the difference between the retrieved memory of the incoming item and the actual item. The item which are no longer relevant, e.g., already recalled item in serial recall task, are actively removed from the memory via anti-learning. The performance is mainly determined by interference within context layer and content layer. Those assumptions also hold in SOB-R. However, SOB-R modifies the assumptions about the precision of accessing the context in SOB-CS.

2.1.1 Shadow of previously used context

SOB-CS assumes that memory system has precise access to context, thus, whenever a context is required during encoding or retrieval, the required context can be accessed without erroneously accessing the wrong context. However, some models do not share this assumption. In OSCAR (Brown, Preece, & Hulme, 2000), the serial-position information might not be successfully reconstructed from the oscillator information, which results in recalling in the wrong serial position. In a free recall model proposed by Farrell & Lelièvre (2012), the serial-position information is retrieved with the help of a grouping cue and a trial cue, and the precision of serial-position information depends on the interference between bindings. The error of retrieving the wrong context information is a source of error in the model. Though SOB-CS assumes that context can be retrieved without confusion, the succeeding SOB-R does not incorporate the same assumption.

SOB-R assumes that the shadow of previously used contexts has influence on the current context, both during the encoding process and the recall process, i.e., whenever a context is used, instead of using the intended context, a mixture of the intended context and the previously used context is used instead. While encoding, instead of forming binding between the current item and the current serial position, the item is bound with the current serial position and the previous serial positions. The same happens during the serial-recall process. When using the serial position to retrieve the item bound to it, instead of using the serial position itself, a mixture of the intended serial position and the previous serial positions is used to retrieve the item. The shadow of previously used contexts predicts different serial-position effects between using the content to retrieve the context and using the context to retrieve the content. Because the first serial position has a continuous influence on the later context, the first serial position is bound to all the items in the memory list. The last serial position, however, is only bound to the last item in the memory list. Thus, when using the serial position to retrieve the item, the precision of correctly retrieving the item bound to the serial position increases monotonically throughout the serial positions. Hence, the model predicts stronger recency gradient compared to primacy gradient in a probed recall task, which is in line with previous findings (Kahana & Caplan, 2002). On the other hand, because there is no interference from shadow of the previous context, the first item is only bound to the first serial position. The last item, however, is bound to all the serial position in the list. Thus the precision decreases monotonically throughout the serial positions when using an item as a cue to retrieve its serial position, which results in stronger primacy gradient and weaker recency gradient. The prediction was confirmed in position recall studies (e.g. Jahnke, Davis, & Bower, 1989)

The shadow does not imply that the precision of context decreases with time. Some models, e.g. SIMPLE (Hay et al., 2007), assumes that the discriminability on the time dimension decreases with time. The ability of discriminating two events that happened at a constant temporal distance decreases with the time between events and the attempt to recall them. However, in SOB-R the precision of context does not change with time or events. The shadow causes the precision of encoding to decrease with every use of context, but the precision of the context itself does not change.

2.1.2 Assumptions about the recognition process

SOB-R assumes that the recognition process comprises comparing the retrieved memory content to the probe. Since SOB-R assumes the memory for recognition tasks as the binding between the context and the content, the context is needed to retrieve the bound content from the memory. However, because the context of the probe is not provided in most of the recognition tasks, the context of the probe has to be retrieved from the memory. To retrieve the context of the probe, regardless of whether the probe appeared in the memoranda or not, the probe is treated as the activation cue for retrieving the context bound to it in the memory. At the beginning of the recognition process, the retrieved context is noisy because of the interference from all the bindings in the memory. SOB-R assumes that the retrieved context goes through a deblurring process similar to the deblurring process for retrieved content in SOB-CS. While the retrieved context goes through the deblurring process, the content is constantly retrieved from the memory and is compared to the probe. At the beginning of the recognition process, retrieved context is a mixture of all the position representations used in the present list, weighted by the similarity of the item in each position to the probe (such that positions with items more similar to the probe receive higher weight). Thus the comparison between retrieved memory and the probe is similar to a global comparison. After the retrieved context is deblurred from the noise, the retrieved memory consists of mostly the content bound to that particular context, which is more similar to a local comparison.

The recognition process is SOB-R is similar to the recognition process in IRM, in which the retrieved information begins from the average of all the items in the memory and gradually narrowing down toward a single item in the memory. However, IRM and SOB-R treat new probes differently. In IRM, because a new probe does not resonate to any item in the memory, the retrieved memory gradually becomes nothing, thus the evidence for rejecting the new probe increases over time. In SOB-R, because the context used to retrieve content is retrieved based on activation of the probe in the content layer, the context with the strongest binding to the probe is retrieved regardless of whether the probe is in the memory list or not. Thus, while the retrieved context is gradually being deblurred, the retrieved memory is gradually narrowed toward the the item most similar to the probe in the memoranda, which results in increasing evidence of accepting the new probe.

SOB-R assumes the energy of the probe is used as the evidence in the recognition process. Since the energy in SOB-CS is measured from the comparison between the retrieved content and the actual item, the energy of the probe comes from the comparison between the retrieved memory and the actual probe. The energy is then compared to a constant threshold. If the energy of the probe is larger than the threshold, meaning that the retrieved memory is similar enough to the probe, the probe is likely to be accepted as old item. If the energy is smaller than the threshold, the retrieved memory is dissimilar to the probe, which leads to rejection of the probe. The threshold for determining the acceptance and rejection of the probe is set to be the average similarity between any individual content item and the average of all content items (i.e. the prototype of content) in the experiment. Previous studies have shown that participants are sensitive to the similarity between items and adjust their decision threshold accordingly (Brainerd, Reyna, & Kneer, 1995; Cleary, Morris, & Langley, 2007; Kahana, Zhou, Geller, & Sekuler, 2007). Thus SOB-R assumes that the prototype of items and the similarity between individual items and the prototype of items can be quickly assessed during the experiment, and the average similarity is used as threshold.

SOB-R uses the Diffusion model (Ratcliff, 1978) as an interface model to convert the evidence to reaction time and proportion of correct responses. Because most recognition tasks measure reaction time and proportion correct as measurement of performance, allowing SOB-R to predict the same measurements makes it is easier to assess the performance of the model.

However, SOB-R only produces evidence in favor of accepting or rejecting the probe. Thus an interface model is needed to convert evidence to reaction time and proportion correct. The interface model of choice is a simplified version of the diffusion model (Wagenmakers, van der Maas, Dolan, & Grasman, 2008). The diffusion model is a measurement model for reaction time and proportion of correct responses for two-alternative choices task. In the diffusion model, an accumulator is constantly accumulating evidence toward one of the two response choices. The response is made when the accumulated evidence reaches one of the two response boundaries, each representing one of the response choices. Depending on the reached boundary, the respond is made accordingly. In SOB-R, the response boundaries are defined as acceptance and rejection, and the energy is used as the mean drift rate for the accumulator. Because the energy of the probe changes over time, the mean drift rate also changes accordingly. It is difficult to archive the analytic solution for the diffusion process when the drift rate changes constantly. Thus SOB-R simulates the diffusion process instead to obtaining the reaction time and proportion correct. Also, in order to avoid over-explaining the data through the interface model, SOB-R simulates all conditions in a task with the same set of parameters of the interface model, thus the difference between conditions is solely explained by the core model instead of by the interface model.

2.2 Content and context representation

In SOB-R, content and context are represented in a distributed fashion. Instead of a single unit being designated to represent a content, a content is represented with several units, where each unit could represent a feature or a group of features. All the contents share the same units with different values of individual units, representing the different feature value in different contents. The similarity between two contents can be measured by calculating the *dot product* between two contents or calculating the *cosine* between two contents. The difference between *dot product* and *cosine* is that *cosine* normalizes the length of vectors, i.e.

the number of nodes, while *dot product* does not. In SOB-CS, the similarity is measured by the *dot product*. In SOB-R, the similarity is calculated though the *cosine*.

Table 1
Summary of the parameters in SOB-R along with the value of the parameters in individual simulation.

Parameters	Brief description	Sternberg's task, extralist-feature effect, the expectation effect-recognition	Speed-accuracy trade-off	Serial-recall task, Local recognition task, the expectation effect-serial recall.	Continuous stimulus
s_v	Similarity between items	0.25	0.25	0.25	3.0
s_p	Similarity between neighboring context	0.5	0.5	0.5	0.5
e	Threshold for the logistic function for converting energy into encoding strength	0.5	0.5	0.5	0.5
g	Gain for the logistic function for converting energy into encoding strength	6.0	6.0	6.0	6.0
R_e	Encoding rate for items	6.0	6.0	6.0	6.0
c	Content distinctiveness during serial recall	-	-	10	-
res_p	Amount of influence from the shadow	0.3	0.3	0.1	0.3
dc	Context deblurring rate	0.2	0.8	0.2	0.2
N_o	Noise in the memory network	0.8	0.8	0.8	0.8
a	Boundary separation in the interface model	0.2	0.2	0.2	0.2
z	Starting point in the interface model	0.1	0.1	0.08	0.1
Ter	Non-decision time in the interface model	0.3	0.3	0.3	0.3
τ	Threshold for accepting or rejecting the probe	0.5	0.5	0.5	0.5
q	Scaling parameter in the interface model	1.5	1.5	1.0	1.5
s	Noise in the interface model	0.1	0.1	0.1	0.1

Content in SOB-R is represented by 120 units. Each unit can have a value of either $+1$ or -1. In order to have a better control over the similarity between the items, all the items are created by changing the elements from a prototype item. The prototype item is created by randomly choosing $+1$ and -1 for individual units with equal probability. To create an individual item, each unit has s_v probability to alternate the units in the prototype item and $1 - s_v$ probability to copy the unit in the prototype item. The expected value of the average similarity between prototype and items is $1 - 2s_v$.

Context is represented by 16 units. Unlike the content, the values in the content are not limited to be $+1$ or -1 but any real number. SOB-R assumes that the similarity between serial positions is reduced monotonically when the distance increases. The similarity between context p_i and p_j is defined as:

$$sim(p_i, p_j) = s_p^{|i-j|}, \tag{1}$$

where the s_p is the similarity between closest neighbor positions and ranges from 0 to 1, and i and j are the serial positions. To create the contexts such that they meet the constraint of Equation 1, a weighted matrix which meets the constraint is created, and Walsh matrix is combined with the weighted matrix. The weight matrix W is constructed according to:

$$W = \begin{bmatrix} 1 & s_p & s_p^2 & \cdots & s_p^{n-1} \\ 0 & \sqrt{1-s_p^2} & s_p\sqrt{1-s_p^2} & \cdots & s_p^{n-2}\sqrt{1-s_p^2} \\ 0 & 0 & \sqrt{1-s_p^2} & \cdots & s_p^{n-3}\sqrt{1-s_p^2} \\ 0 & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \sqrt{1-s_p^2} \end{bmatrix}. \tag{2}$$

Each column in the Equation 2 follows the same constraint as context as in Equation 1, i.e., the cosine between column i and j is $s_p^{|i-j|}$. The columns in the Equation 2 serve as the weighting in the weighted sum of the rows in the Walsh matrix in order to create corresponding contexts, which follows:

$$p_i = w_i \cdot H(16) \tag{3}$$

where the w_i is the vector of the ith column in the weight matrix W, and the $H(16)$ is the Walsh matrix with length 16. Because all rows in the Walsh matrix are orthogonal to each other, the cosines between context vectors retain the same value as the cosines between columns of the weight matrix W. Since W follows the constraint in Equation 1, the created contexts also follow the same constraint in Equation 1.

2.3 Encoding process

The encoding process of SOB-R consists of forming the binding between the item and the serial position. The binding is formed via Hebbian learning (Hebb, 1949), which associates features of item v_i and serial position p_i. The learned association is then superimposed on the memory trace C. However, because SOB-R assumes that the previously used context has influence on the usage of current context, instead of binding v_i with pure p_i, binding is formed between v_i and $p(i)$ where

$$p(i) = p(i-1) \cdot res_p + p_i \cdot (1 - res_p). \tag{4}$$

The current used context $p(i)$ is a weighted average between the last used context $p(i-1)$ and the intended serial position p_i. The res_p is the weighting between the previous used context and intended serial position. Smaller res_p indicates the used context is less affected by the previous context.

The change of the memory trace after encoded item i is:

$$\Delta C_i = \eta_e(i) v_i p(i)^T. \tag{5}$$

The encoding strength $\eta_e(i)$ of item i is a factor of the available time for encoding the item, t_e through:

$$\eta_e(i) = A(i)(1 - \exp(-t_e R)), \tag{6}$$

where the $A(i)$ is the asymptote of encoding strength, which is the maximum encoding strength given infinite time of encoding. R is the rate of encoding that modulates the speed of the encoding strength reaching the asymptote. The asymptote of the encoding strength, $A(i)$, is determined by the energy of the item, which is the similarity between the retrieved expectation about the item and the item itself, through a logistic function as:

$$A(i) = \frac{1}{1+\exp(-(-E_i-e)g)}. \tag{7}$$

the E_i is the energy of the item i, and e and g are the threshold and the gain parameters, respectively. Since the memory trace C is the association between content and context, the content can be retrieved by activating the context via $C_{i-1}p(i)$, and the energy of item i is

$$E_i = cos(v_i, C_{i-1}p(i)). \tag{8}$$

When the energy is smaller[1], i.e. the retrieved memory and the item are more dissimilar; the asymptote of the encoding strength becomes larger. Thus by giving the same amount of time for encoding, smaller energy will result in stronger encoding strength. At the beginning of the encoding process, since there is no item in the memory, thus energy is low because only random noise and residual of previous trials are retrieved from the memory trace, which results in larger encoding strength. The energy gradually increases when more items are encoded in the memory. Because the binding between context and content superimposed on memory trace, the retrieved expectation gradually becomes similar to the prototype of the content when more bindings are superimposed in the memory trace. The encoding strength gradually decreases correspondingly. The energy and the correspond encoding strength are shown in Figure *4*.

[1] The energy in the SOB-CS and SOB-R have different notion. In SOB-CS, the energy is the negative of the dot product between the expectation and the to-be-encoded item. Thus in SOB-CS the smaller the energy is, the more similar the expectation and the to-be-encoded item are. In SOB-R, however, the energy is calculated by the cosine of the expectation and the to-be-encoded item. Thus the smaller the energy is, the more dissimilar the expectation and the to-be-encoded item are.

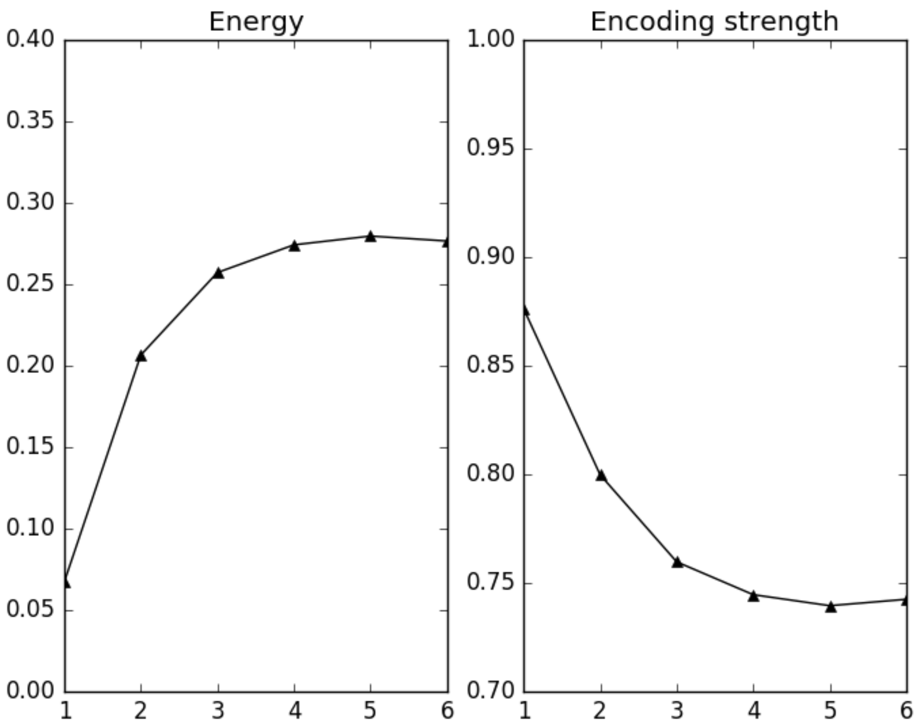

Figure 5. The energy and the encoding strength across serial positions. The energy and the encoding strength are simulated with the parameters set of Sternberg's task listed in Table 1.

2.4 Serial-recall process

The recall process in SOB-R is the same as the recall process in SOB-CS. For the serial-recall process it is assumed that the serial position is used as a cue to retrieve the bound item. Because of the overlap between contexts, the retrieved item is noisy. The noise in the retrieved item is then removed via a deblurring process, and the deblurred item is recalled. SOB-R also assumes response suppression: the recalled item is removed from the memory trace to avoid recalling the same item. After each response, the memory trace suffers from output interference, where a random noise is added to the memory trace.

Retrieving content from memory trace is the same as retrieving an expectation. As during the encoding process, the context used to retrieve content is affected by the previous used context. The usage of context is interfered with by the shadow, thus instead of using the context representing serial position j, p_j, to retrieve the content, a blended context $p(j)$ is used as in Equation 17. The retrieval process is

$$v_j' = C_i p(j) \tag{9}$$

to retrieve the content v_j'. Because the overlapping between serial positions, the retrieved memory v_j' contains the item bound with serial position j and the other items in the neighboring positions. To remove the noise in v_j', the retrieved memory is processed through a dynamic iterative deblurring process. In practice, instead of processing though the deblurring process, SOB-R simulates the result of the deblurring process by calculating the similarity between retrieved memory and all the possible response candidates. The probability of recalling a certain response candidate v_l is

$$P(v_l) = \frac{\mathrm{e}^{cos\left(v_j', v_l\right) \cdot c}}{\sum_i^n \mathrm{e}^{cos\left(v_j', v_i\right) \cdot c}}, \tag{10}$$

where c represents the distinctiveness between each response tendency and ranges from zero to infinity. When c is zero, all the candidates are equally plausible to be recalled, regardless of the similarity between retrieved item and the response candidates. When c increases, the distinctiveness between response candidates also increases, which results in the most matching candidate being selected more frequently.

The recalled candidate is then suppressed from the memory in order to avoid repetition. SOB-R assumes that response suppression is implemented by anti-learning. Thus the response suppression is the same as Equation 5 with negative encoding strength, and the change of memory trace after response suppression is

$$\Delta C_j = \eta_s(j) v_{o,j} p(j)^T. \tag{11}$$

The learning rate $\eta_s(j)$ varies based on the energy ratio between the energy for the first recalled candidate and the energy of the current recalled candidate as

$$\eta_s(j) = -\frac{E_j}{\phi_s E_1}, \tag{12}$$

where the ϕ_s is the scalar to adjust the ratio between the energy of the first recalled candidate and the energy of the current recalled candidate. The anti-learning rate of the first recalled candidate is $-1/\phi_s$.

After every response, SOB-R assumes the response creates output interference in the memory trace. The output interference is implemented as adding a Gaussian noise, N_o, to the memory trace C.

2.4 1 Simulation of serial-recall task

Since SOB-R alternates the accessibility of context from SOB-CS, to ensure that SOB-R still retains the ability to simulate the serial-recall task, I compare the simulation of serial-recall task from SOB-CS and SOB-R, before continuing to introduce the recognition process of SOB-R. The parameters used to simulate the serial-recall task are listed in Table 1 for serial recall. The simulation from SOB-CS is shown on the left side of Figure 6, and the simulation from SOB-R is shown on the right side of Figure 6. Generally, the simulation from SOB-CS and SOB-R show similar serial-position effects with a strong primacy gradient and a weak recency gradient. Also, the simulation from SOB-R shows stronger recency effect. Overall, SOB-R can still simulate the serial-recall task, even with the modifications.

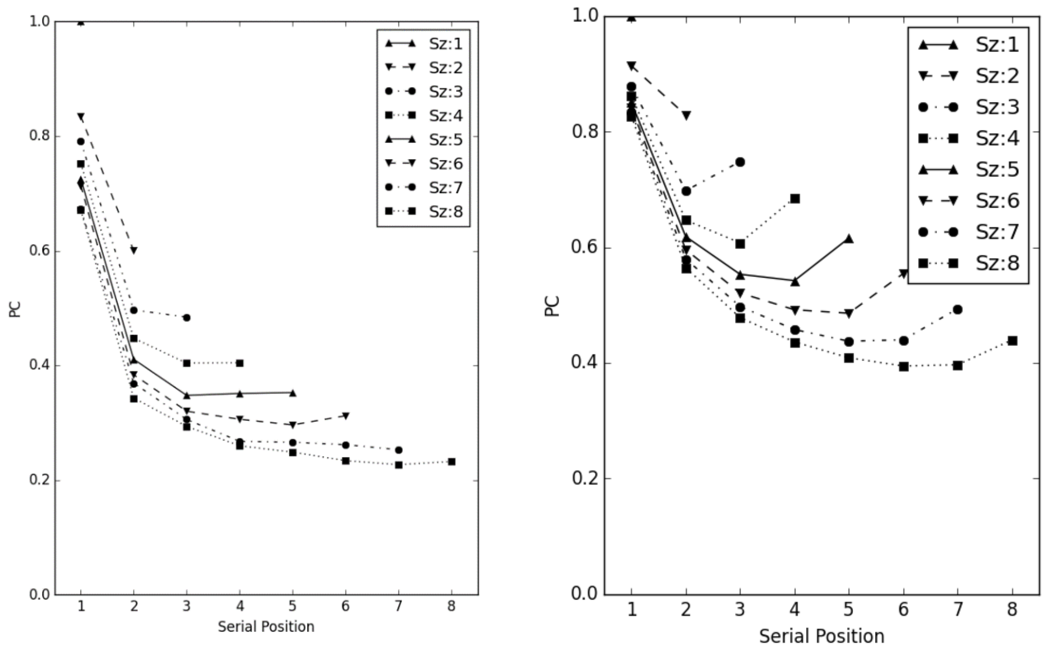

Figure 6. The simulated serial-position effect from SOB-CS (left) and SOB-R (right). The parameters used to simulate the serial-position effect is listed in Table 1.

2.5 Recognition process

I assume that the recognition process in SOB-R is comparing the retrieved content to the probe. To retrieve content from the memory trace, a context is retrieved by using the probe as cue. Because the overlap between bindings, the retrieved context is noisy and is deblurred via the delurring process in the same way as the deblurring process of noisy content. While the context is undergoing the deblurring process, the content is constantly retrieved and compared to the probe. Thus, the result of the comparison changes over time. The result of comparison is used as the evidence for accepting or rejecting the probe, which is fed to an interface model to determine the reaction time and the response.

To retrieve the context of a probe (p_p'), the probe (v_p) is used as cue to activate the context through the memory trace via

$$p_p' = v_p C. \tag{13}$$

The retrieved context p'_p is then deblurred though the same deblurring process as is used for deblurring content. In practice, SOB-R utilizes a similar simplified deblurring process as SOB-CS. For each iteration of the deblurring process, the deblurred context at the *t*th iteration $p_p(t)$ is the weighted sum of all contexts used in the trial

$$p_p(t) = \sum w_i(t) \cdot p_i, \tag{14}$$

where the weighting $w_i(t)$ is based on the similarity between the retrieved context and each context used in the trial through

$$w_i(t) = \frac{\mathrm{e}^{sim(p'_p, p_i) \cdot c(t)}}{\sum_j^n e^{sim(p'_p, p_j) \cdot c(t)}}. \tag{15}$$

The distinctiveness parameter $c(t)$ begins from zero and increases linearly with each iteration though

$$c(t) = dc \times (t - 1). \tag{16}$$

At the beginning of the deblurring process, the distinctiveness parameter $c(t)$ is zero, thus the weighting of each context $w_i(0)$ become $1/n$. The deblurring context is the average of all the contexts in the trial. Later in the deblurring process, the distinctiveness becomes larger to distinguish the context most similar to the retrieved context from the other contexts in the trial; the deblurred context becomes the context most similar to the context retrieved from the probe.

While the retrieved context is undergoing deblurring, the context is used as a cue to retrieve the content bound to it from the memory trace. However, because the shadow of the previously used context always has influence on the current context, the context used to retrieve the content is no exception. Thus, $p(p_t)$ is used as cue instead of $p_p(t)$ with Equation 17, as

$$p(p_t) = p(n) \cdot res_p + p_p(t) \cdot (1 - res_p), \tag{17}$$

where the $p(n)$ is the shadow after encoded the last (nth) item. The retrieved memory at iteration t is

$$v'_{p,t} = C_p p(p_t), \tag{18}$$

and the result of comparison between the retrieved memory and the probe, i.e. energy, is

$$E_p(t) = \cos(v'_{p,t}, v_p). \tag{19}$$

Because the noisy context is gradually sharpening toward the position marker of the item that is most similar to the probe, the items other than the one bound to that context are taking less weight in the retrieved memory. Thus the energy of the probe increases with the iterations of the deblurring process, as shown in Figure 7. It is important to note that even though a new probe was not bound to any context in the memory, the energy of a new probe still increases throughout the deblurring process. This is because when using the probe as cue to retrieve the context, the context which is bound to the item most similar to the probe is retrieved. Therefore, throughout the deblurrring process, the retrieved memory is gradually sharpened toward the item in the memory that is most similar to the probe, which results in increasing of energy. Even though the energy of a new probe increases with the deblurring process, the item retrieved from the sharpened context is still different from the probe, thus the energy of new probes does not increase as much as the energy of positive probes.

The energy of the probe is assumed as the evidence of the probe's presence in the memory. If the evidence shows that the probe has strong presence in the memory, the probe will be accepted as positive probe. On the other hand, if the evidence shows low presence of probe in the memory, the probe will be rejected. The threshold for accepting or rejecting the probe is assumed to be the average similarity between all the items and the prototype of items, i.e. $1 - 2s_v$. The threshold is assumed to be constant and does not change with set size or serial position. Thus the set-size effect and the serial-position effect are purely affected by the strength of evidence instead of different thresholds.

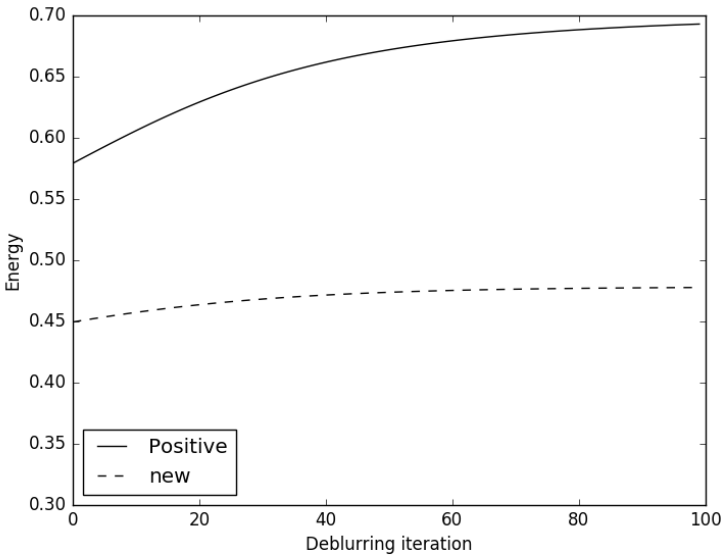

Figure 7. The change of energy through the iteration cycle. The energy is simulated with the parameters set of Sternberg's task listed in Table 1.

To convert the strength of evidence to reaction time and proportion correct, a simplified version of the diffusion model is used as an interface model. Unlike the full diffusion model, the simplified diffusion model assumes no variability of non-decision time and starting point of accumulator between trials. The upper decision boundary is defined as accepting the probe, and the lower decision boundary is defined as rejecting the probe. The drift rate of the accumulator is the difference between the energy of the probe and the threshold, multiplied with a constant scalar q. The probe is easier to be accepted when the evidence for its presence in the memory is stronger, and the probe is most likely to be rejected when the evidence for its presence is weaker. Thus the absolute difference between energy and threshold reflects the easiness of accepting or rejecting the probe, which is translated in the drift rate of the accumulator in the diffusion model. Because the energy of probe changes with the deblurring process, the drift rate also changes accordingly. Thus the drift rate at the *t*th iteration of deblurring process $\nu(t)$ is

$$\nu(t) = \left[E_p(t) - \tau\right] \cdot q, \tag{20}$$

where τ is the threshold of accepting or rejecting the probe, and q is the scalar for the drift rate.

Because the drift rate changes over time, and the change of drift rate is dependent on the probe and the memory trace, it is difficult to acquire the analytic solution for the diffusion model. Instead, the diffusion process is simulated as a random walk process with small step size to obtain the mean reaction time and the proportion of accepting and rejecting the probe. The step size of simulation was 5ms per time step, and each trial was simulated 100 iterations to obtain the mean reaction time and the proportion of responses.

Because the diffusion model is flexible enough to simulate many effects by assigning different parameters to different conditions, to ensure that the core model is responsible for the simulated effect, the parameters in the diffusion model are constant within the experiment. Thus the simulated effect is contributed by the core model instead of interface model.

2.5.1 Simulation of Sternberg's task

The Sternberg task is the most well know paradigm of short-term recognition. In Sternberg's task, the participant is presented with several memory items sequentially. After all the items have been presented, a probe appears on the screen. The participant is asked to judge if the probe is among the memoranda as fast as possible. The typical finding with Sternberg's task is that participants are fairly accurate with their response. The reaction time for accepting the positive probe (the probe in the memory list) and the reaction time of rejecting the new probe (the probe not in the memory list) increases linearly and in parallel with set size, as shown in Figure 8. Serial-position effect is also found consistently and shows strong recency gradient, the reaction time is fastest at the end of the list and slower toward the middle of the list, and there is a weak primacy gradient, the reaction time is slightly faster at the beginning of the serial position, as show in Figure 11.

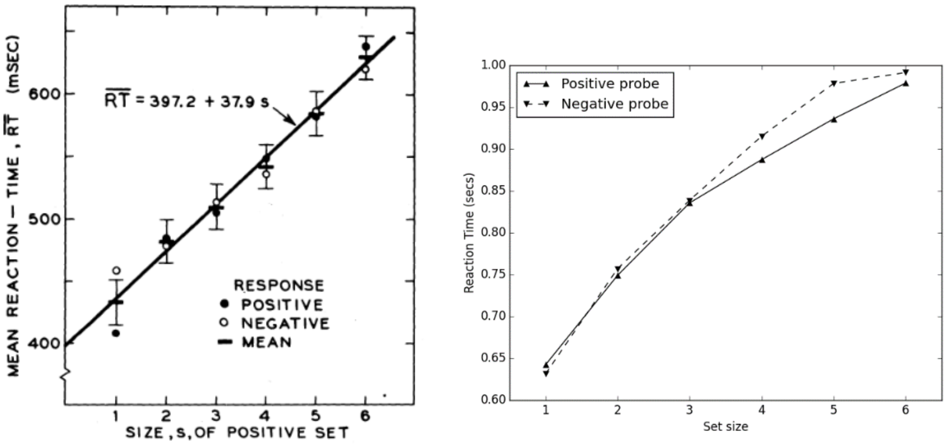

Figure 8. The observed set-size effect (left) and the simulated set-size effect (right). The observed set-size effect is the Figure 4 of Sternberg (1969). The simulated set-size effect is simulated with the parameters set of Sternberg's task listed in Table 1.

The parameters for simulating the Sternberg task is listed in Table 1. The simulation assumes no carry over between trials. The memory trace resets after each trial, and there is no influence from the context used prior to the trial. The simulated set-size effect is shown in Figure 8. Though the simulated reaction time increases monotonically with set size, and the reaction time for positive probe and new probe increase in parallel, the reaction time does not increase linearly. Instead, the increase of the set-size effect decelerates. SOB-R simulates the set-size effect by the interference between bindings. When more bindings are encoded into the memory, the amount of crosstalk between bindings also increases, which results in retrieving more items when activating the context. The positive probe becomes harder to be accepted because the similarity between probe and the retrieves content is distorted by the other items. Also, because more items are contributed retrieved to the retrieved content, the retrieved memory becomes similar to the average of items, i.e. the prototype of items, which results in increasing similarity between new probes and the retrieved memory. However, the amount of crosstalk between bindings does not increase linearly with set size. Instead, the amount of crosstalk increases with deceleration because of the constraint in Equation 1. Thus SOB-R simulates the decelerated increasing of the set-size effect.

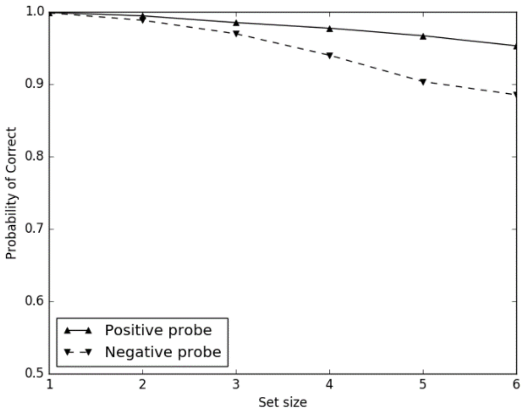

Figure 9. The simulate set-size effect of accuracy. The simulated set-size effect is simulated with the parameters set of Sternberg's task listed in Table 1.

The simulated set-size effect of accuracy is shown in Figure 9. The accuracy of both positive probes and the new probes decrease when the set size increases. In most of the Sternberg task, the accuracy is mostly at ceiling (Sternberg, 1969). The drop of accuracy in larger set sizes is the result of the parameters constraint in the interface model. Because the parameters are fixed across different set sizes, the increasing of reaction time will result in the decreasing of accuracy.

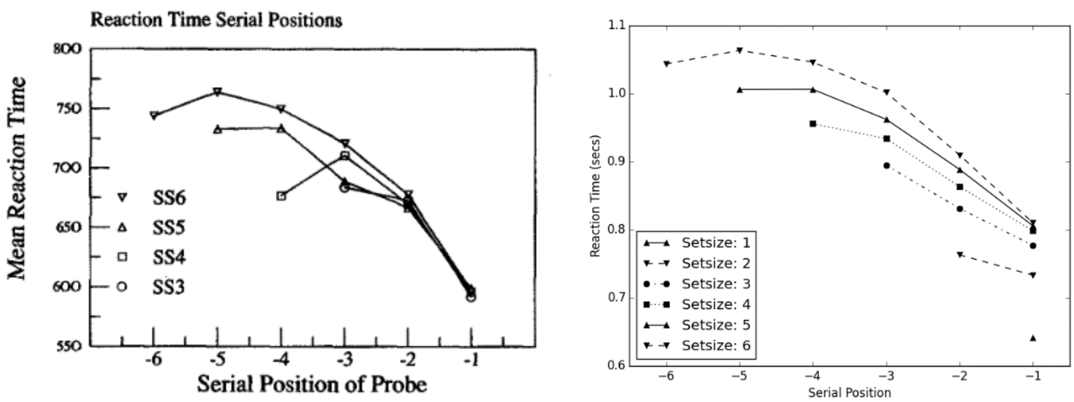

Figure 10. The observed serial-position effect (left) and the simulated serial-position effect (right). The observed serial-position effect is from the Figure 9 of McElree & Dosher (1989). The simulated serial-position effect is simulated with the parameters set of Sternberg's task listed in Table 1.

The serial-position effect simulated by SOB-R is shown in Figure 11. The simulated serial position effect has a strong recency gradient and a weak primacy gradient, the same as previous experiment findings. Previous experiments also have shown that the most recent item is responded with fastest reaction time, and the reaction time is the same regardless the set size (McElree & Dosher, 1989) or show the least amount of set-size effect (Nosofsky et al., 2011). The simulated results of SOB-R are similar to the empirical findings. Although the reaction time of the most recent item still increases with set size, the amount of increase is the smallest compared to the rest of serial positions.

SOB-R simulates the recency gradient though the shadow of previous used context. Because the current context is affected by the context used previously, when using context to retrieve the content in Equation 18, the shadow of recently used context is blended into the retrieval context. Since the recently used contexts are the serial positions near the end of list, the retrieval context is shifted toward the recent serial positions. Thus the retrieved content is also shifted toward the recent items in the memory list. The increasing proportion of recent items in the retrieved memory has benefit for the items near the end of the list and disadvantage for the items near the beginning of the list. Hence SOB-R simulates a strong recency gradient.

The weak primacy gradient is simulated by the primacy gradient of encoding strength as shown in Figure 4. Because the earlier items have stronger binding compared to the other items, the earlier items are retrieved more strongly by activation of the corresponding context, which results in a primacy gradient in the serial-position curve. The primacy gradient is not as strong as the recency gradient because of the shadow of previously used contexts. Since the shadow of previous contexts shifts the retrieved memory toward the items near the end of the list, the strength of the primacy gradient is reduced.

In the SOB-R simulation, the last item showed the least effect from set size. This is because of the shadow of previous used context. The influence comes from both encoding and recognition process. During the encoding process, as mentioned previously, the last serial position is only bound to the last item in the list, unlike the other serial positions, which are bound to more than one item. Therefore, cueing with the last serial position suffers the least amount of crosstalk from other serial positions. During the recognition process, the context used to retrieve the content is also affected by the shadow of previously used context. Since the last used context during the encoding process is the last serial position, the shadow of previously used context during the recognition process has the least amount of distortion on the probe from the last serial position comparing to the probe from other serial positions, because the shadow is almost identical to the to-be-retrieved context, so that the retrieved context has the least amount of interference from other serial positions. Because of the combination of the least amount of crosstalk between bindings and the least amount of interference from the shadow of context, the last serial position has the least effect from set size.

Simulating other tasks

3.1 Speed-accuracy trade-off

To simulate the speed-accuracy trade-off function, SOB-R assumes the response deadline only affects the decision making process and has no effect on the encoding and the recognition process, thus the core model is not changed at all, but the interface model is adjusted to simulate the speed-accuracy trade-off function. SOB-R assumes the response is determined by the amount of evidence in the accumulator when the response deadline is reached. If the accumulator has accumulated more evidence toward accepting the probe compared to the starting point, then the response made at the response deadline is accepting the probe, and vice versa. If the accumulator has accumulated the same amount of evidence as the starting point, the response is made randomly. The response is only made at the response deadline. The accumulator will continue accumulating evidence even when the response boundary is reached, and the response is determined by the amount of evidence at the response deadline regardless of whether the response boundary has been reached or not.

The simulated set-size effect of the speed-accuracy trade-off function is shown in Figure 11. The simulated speed-accuracy trade-off function is similar to the observed speed-accuracy trade-off function. Smaller set sizes have higher increasing rate and asymptote compared to larger set sizes. SOB-R simulates the increasing rate and the asymptote by different mechanisms. The increasing rate is determined by the context deblurring, and the asymptote is determined by the amount of interference generated by the context overlapping. The context deblurring affects the retrieved content, which in turns affects the drift rate in the interface model. Thus, the speed of deblurring process affects the speed with which the drift rate reaches the maximum, which in turn affects the increasing rate in the speed-accuracy trade-off function. With smaller set size, the noise contained in the retrieved context is also smaller since there are less bindings encoded in the memory, thus it takes less time to deblur the context. The simulated asymptote is affected by the amount of interference from non-target items in the retrieved memory after the context finished deblurring. Since asymptote is not affected by the interval between the probe onset and the response deadline, it is safe to assume that the recognition process reached a stable state after the speed-accuracy trade-off reached the asymptote. In the core model of SOB-R, context deblurring is the only process affected by time in the recognition process. Because the recognition process has reached a stable state when the speed-accuracy trade-off function reached asymptote, it is reasonable to assume that the deblurring process has finished at that point. Thus, the asymptote is determined by the final state of retrieved memory, which is affected by the amount of context overlap. In smaller set sizes, the amount of interference generated from the context overlap is smaller compared to larger set sizes, which results in higher asymptote in the speed-accuracy trade-off function.

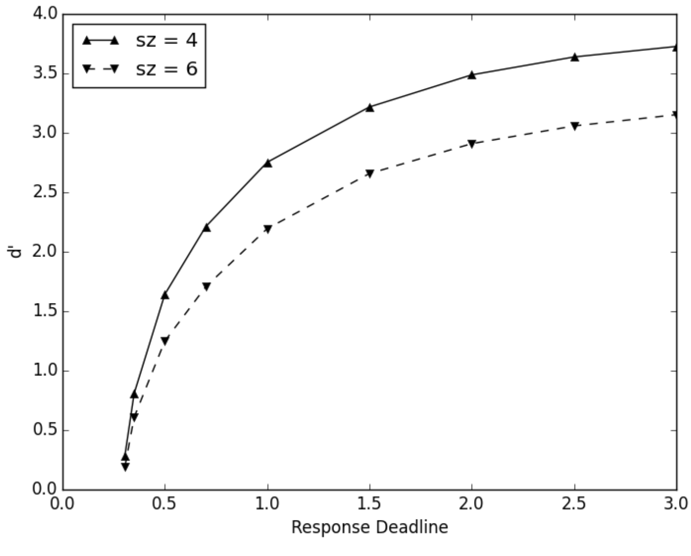

Figure 11. The speed-accuracy trade-off functions for different set sizes simulated by SOB-R. The speed-accuracy trade-off functions are simulated with the parameters set of speed-accuracy trade-off function listed in Table 1.

The simulated serial-position effect of the speed-accuracy trade-off function is shown in Figure 12. Similar to the observed data, SOB-R simulates the probes from the recent serial positions with higher increasing rate and asymptote compared to the probes from the earlier serial positions. SOB-R simulates the higher increasing rate for the probes from the recent serial positions because the shadow of previous used context pushes the noisy context toward the recent serial positions. Though the shadow does not increase the time required for the deblurring process, the shadow distorts the noisy context toward the recent serial positions, which effectively superimposes the correct context to the noisy context and results in faster evidence accumulating. The higher asymptote for the probes from the recent serial positions is also simulated by the shadow in SOB-R. The probes from the recent serial positions suffer less interference from the shadow compared to the probes from the earlier serial position, since the probes matching the items in the recent positions were bound to the recent serial positions. Thus, the amount of interference generated from the shadow is smaller, which results in higher asymptote in the speed-accuracy trade-off function.

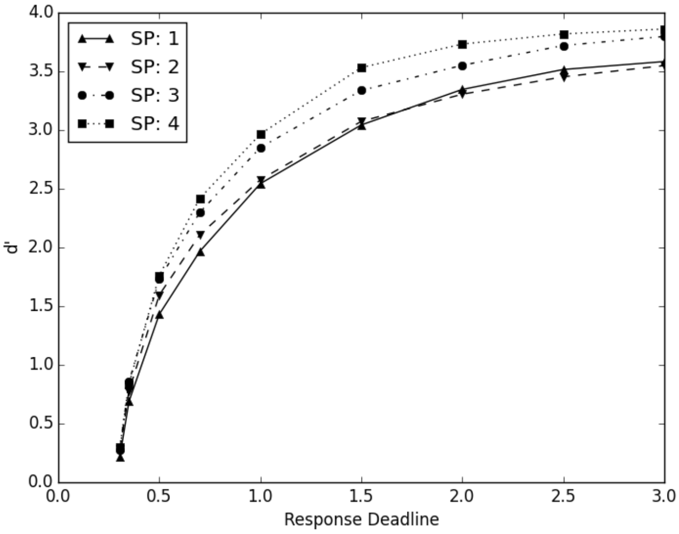

Figure 12. The speed-accuracy trade-off functions for different serial positions simulated by SOB-R. The simulated speed-accuracy trade-off functions for different serial positions are simulated with the parameters set of speed-accuracy trade-off function listed in Table 1.

3.2 Local recognition task

The local recognition task is commonly used to investigate the relationship between local match and global match in recognition process, and to explore the capacity limit in visual working memory. In contrast to Sternberg's task, which only requires participant to judge if the probe is in memoranda or not, local recognition task requires participant to judge if the probe matches the memory item presented at the location of probe. The context information, the position of the item, is important in local recognition task. Since context is an essential part of memory representation in SOB-R, a task that requires context information is a direct test for SOB-R.

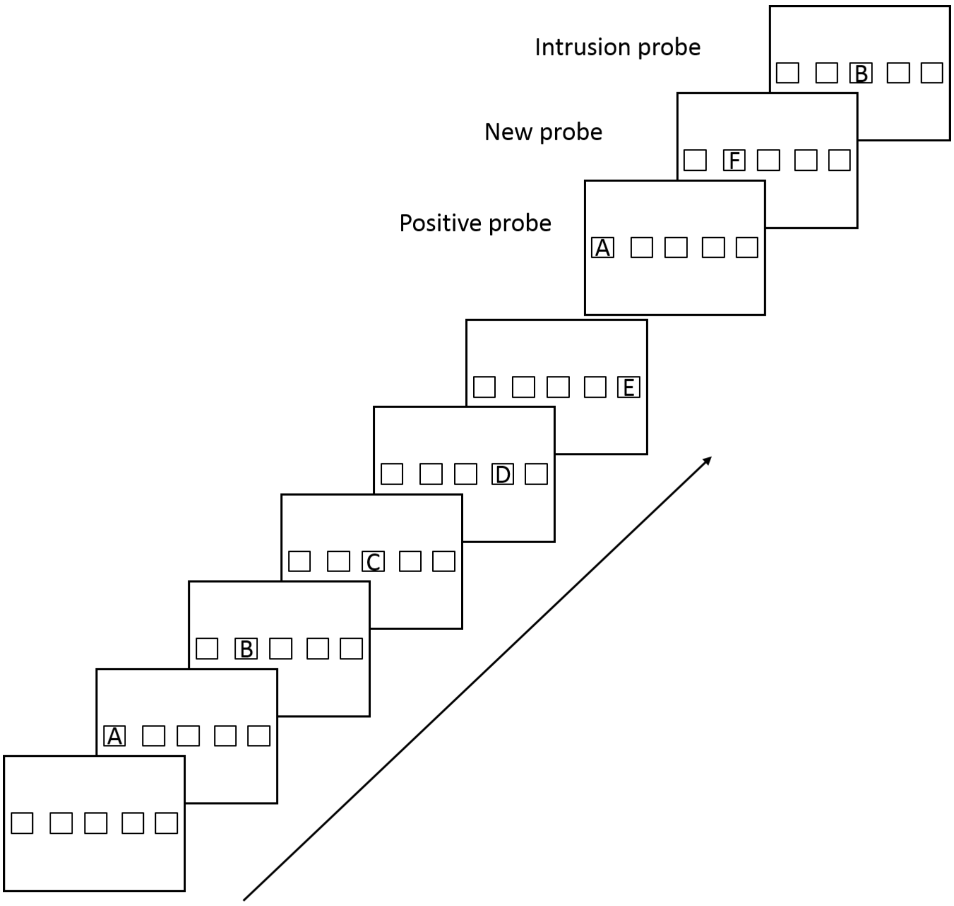

Figure 13. The typical procedure of the local recognition task.

The procedure of the local recognition task is shown in Figure 13. At the beginning of each trial, a small number of empty frames are presented on the screen. The locations of the frames indicate the locations where the items will appear. After the initial interval, the items are presented in the frames sequentially from left to right. After all the items were presented, a probe is presented in one of the frame. Participant has to judge if the probe is the same as the item presented in the same frame. There are three types of probe in the local recognition task:

positive probe, new probe, and intrusion probe. For positive probe, the content of the probe matches the item presented in the same frame. Participant should response “yes” to a positive probe. New probe is defined as that the content of the probe does not match any of the items in the memoranda, thus participant should response “no” to a new probe. An intrusion probe is a probe that matches one item in the memoranda but is presented in the wrong frame. Since an intrusion probe does not match the item presented in the frame in which the probe is shown, participant should response “no” to an intrusion probe.

Previous studies have shown that intrusion probes are more difficult to be rejected compared to new probes, this difference is called intrusion cost (Oberauer, 2003, 2008). A new probe can be rejected through a global match process because of its absence in the memoranda, but an intrusion probe cannot be rejected in the same manner. The content of intrusion probe appeared in the memoranda; hence the global match would suggest a response “yes” to an intrusion probe. There are two possible methods to successfully reject intrusion probes. An intrusion probe can be rejected by comparing the content of the intrusion probe to the item that appeared at the probed location, or by comparing the probed location to the original location of the content of the probe. The earlier method requires retrieving the item information at the probed location, and the later method requires retrieving to which position the probe was bound in the memory. Both methods require access to a specific subset of information in the memoranda; therefore the local match is required to successfully reject the intrusion probe.

Unlike the serial-position effect in the Sternberg task, which can only be measured in the positive probes, in local recognition the new probe and the intrusion probe are presented in a specific location, and therefore the serial-position effect of new probes and intrusion probes can also be measured. Previous studies have shown that the performance is better at the beginning and the end of the serial positions (Oberauer, 2003, 2008). The performance is worse in the middle of the serial positions regardless of the type of probes, as shown in Figure 14. The performance of intrusion probes can also be aggregated according to the position of origin, the position where the content of intrusion probe was originally presented. The performance is better if the origin of the intrusion probe is at the beginning or the end of list. The performance is worse if the intrusion probe originates from the middle of the list, as shown in Figure 15. Though the performance of positive probes can also be aggregated according to the position of origin, the position of origin is the same as the serial position by the definition of positive probe. Thus the performance of positive probe is not shown in the Figure 15. The performance of positive probe and intrusion probe can also be aggregated by the spatial distance between the position of origin and the position of probe. For positive probes, the distance is zero because the position of origin and the position of probe is identical by definition. For intrusion probes, the distance is either larger or smaller than zero according to the difference between the position of origin and the position of probe. The probability of accepting the probe decreases when the spatial distance increases, as shown in Figure 16. Previous research also found the set-effect for intrusion probes (Donkin, Tran, & Pelley, 2014). The performance on intrusion probes decreases monotonically with increasing set size, as shown in Figure 9.

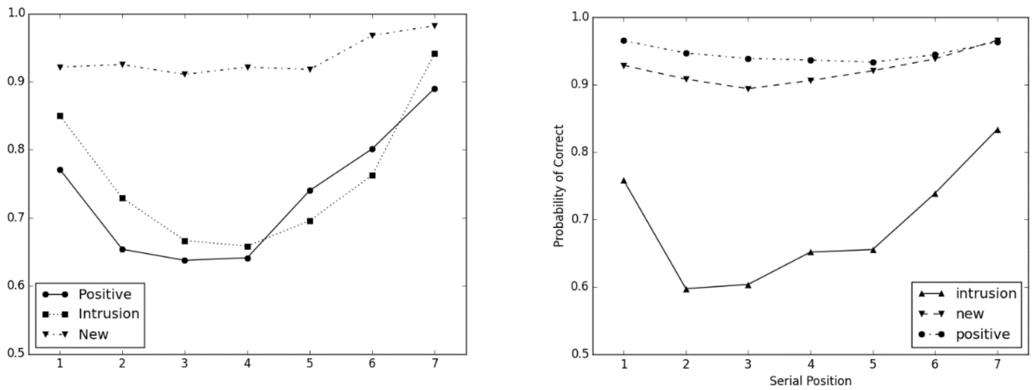

Figure 14. The left figure shows the observed serial-position effect in the local recognition task, and the right figure shows the simulated serial-position effect from SOB-R. The observed data is from an unpublished experiment done in 2014. The simulated serial-position effect is simulated with the parameters set of local

recognition task listed in Table 1.

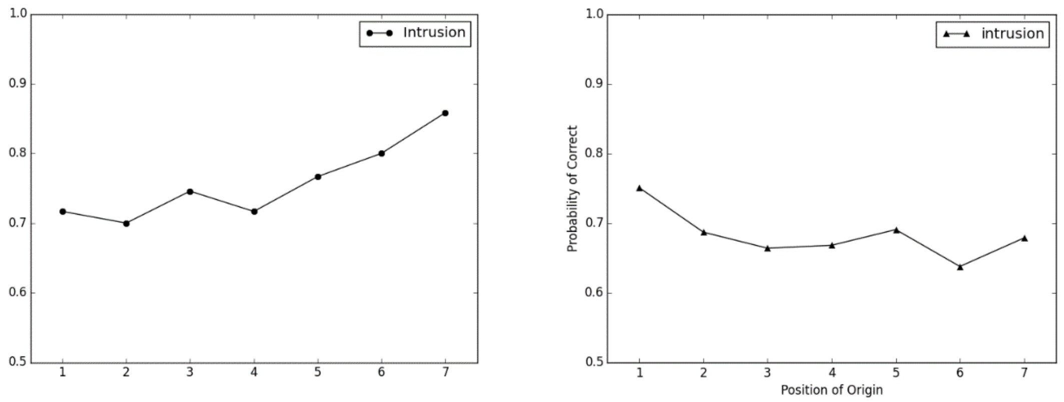

Figure 15. The left figure shows the performance across the position of origin local recognition task, and the right figure shows the simulated performance across the position of origin from SOB-R. The observed data is from an unpublished experiment done in 2014. The simulated serial-position effect is simulated with the parameters set of local recognition task listed in Table 1.

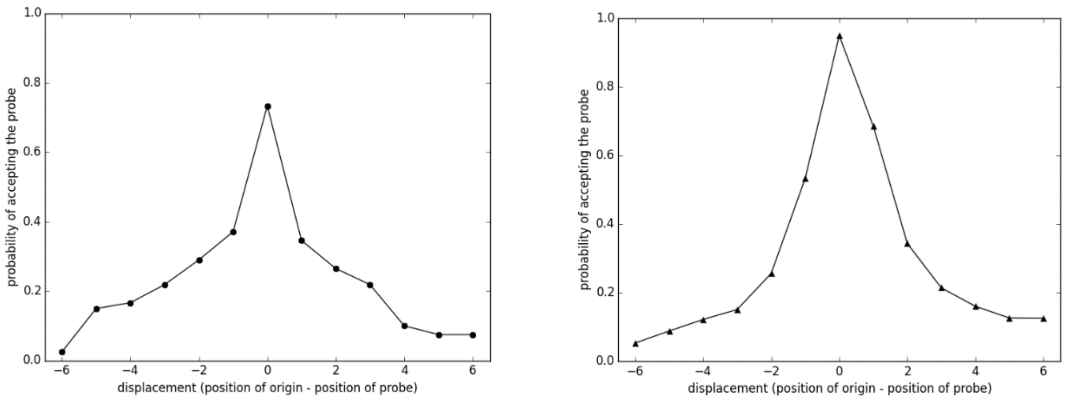

Figure 16. The left figure shows the data of probability of accepting the probe across the distance between the position of origin and the position of probe of local recognition task. The right figure shows the simulated result from SOB-R. The left figure is from an unpublished experiment of local recognition task. The right figure is simulated with the parameters set of local recognition listed in Table 1.

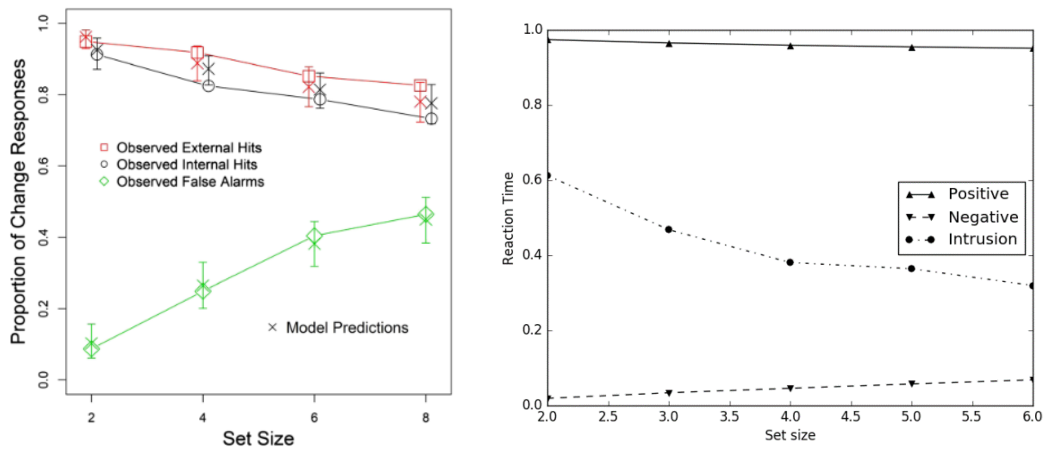

Figure 17. The left figure shows the data of the set-size effect of local recognition task. The Y axis is the probability of rejecting the probe. The red line is the performance for new probe, the black line is the performance of intrusion probe, and the green line is the performance of the positive probe. The right figure shows the simulated result from SOB-R. The left figure is from Figure 3 of Donkin, Tran, & Pelley (2014). The simulated serial-position effect is simulated with the parameters set of local recognition listed in Table 1.

The encoding process of the local recognition task is assumed as binding the item and its location in SOB-R. The items of local recognition task are represented in the same way as the representations of the Sternberg task. The representation of the location follows the same principle as the representation of the serial position in the Sternberg's task. However, similarity between the locations is determined by the spatial distance between locations. Because in the typical local recognition task, the locations of items are arranged either from top to bottom or from left to right, as show in Figure 13, the similarity between locations follows the same constraint as the similarity between serial orders in Equation 1.

For the recognition process, retrieving the context of probe is not necessary in the local recognition task. In the local recognition task, the context of the probe is provided during the recognition process. Thus the context of the probe does not have to be retrieved by using the content of the probe as a retrieval cue. Also, because the context does not have to be retrieved from the memory, the context does not contain noise from the retrieval process, thus the context does not have to be deblurred. Because the context does not change over time through the deblurring process, the retrieved memory does not change with time. The comparison between the retrieved memory and the probe does not change either. Thus the evidence in the local recognition is constant during the recognition process.

The parameters used in the simulation are listed in Table 1. The simulation assumes no carry over between trials, and the memory trace is reset after every trial. The simulation shows that for positive probes and intrusion probes, the probability of accepting the probe decreases as the distance between the position of origin and the position of probe increases, which is consist with the observed data, as shown in Figure 16. SOB-R simulate the result through the amount of context overlap. Because the context overlaps the most with the closest neighbors and less with the far neighbors, the retrieved content contains more items from the closest neighboring positions and less items from the farther positions. As a result, the similarity between the retrieved memory and the memory items gradually decrease when the distance between position of probe and position of items increase. The intrusion cost is simulated through context overlap. When retrieving the expected item at the probed location, the items at neighboring locations are also retrieved. The retrieved content does not only contain the item presented at probed location but also the items from the other locations, which includes the intrusion probe. Thus the intrusion probe is more difficult to be rejected comparing to the new probe.

The simulated serial-position effect is shown in Figure 14. The positive probe, new probe, and intrusion probe show U-shaped serial-position effect where the performance is better at the beginning and the end of the list. The performance of intrusion probe is worse than positive probe and new probe. For positive probe and new probe, the U-shaped serial-position effect is the result of edge effect. Items at the beginning and the end of the list only have interference from one direction, which results in better performance. SOB-R predicts U-shaped serial-position effect for intrusion probe because of the edge effect. For the intrusion probes presented at the edge of list, the retrieved memory only suffers the interference from one direction, thus the retrieved memory consists to a higher proportion of the item presented at the probed location during the encoding. As a result, the intrusion probe is easier to reject, because the item presented at the probed location during the encoding is different from the intrusion probe. The intrusion probes presented at the beginning and the end of the list have lower probability that the probe content originated from the closest neighboring positions comparing to the intrusion probes presented at the middle of the list. Therefore, the average distance between position of origin and the position of probe is larger at the edge of the serial position and smaller at the middle of the list, as shown in Figure 18. Combining with the amount if intrusion cost is affected by the distance between the position of origin and the position of probe, the items from the beginning and the end of the list have better performance than the items from the middle of the list.

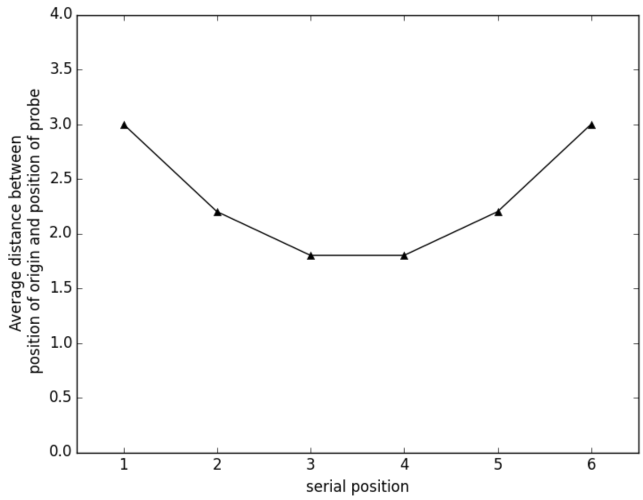

Figure 18. The average distance between position of origin and the position of probe for intrusion probe at different serial positions.

SOB-R also simulates the U-shaped curve of performance over the position of origin for intrusion probes, as shown in Figure 15. Similar to the effect observed from the studies, the performance of intrusion probes is better if the intrusion probes are originating from the beginning and the end of the list. The reason that SOB-R predicts U-shape performance across position of origin is because when the intrusion probe originates from the edge of the list it has a higher chance to be tested at the other end of the list. Since the position of origin and the position of test are separated at the maximum distance, the intrusion cost is at minimum. Thus, the performance at the beginning of the list and the end of the list is better on average. The intrusion probes that originate from the middle of the list have higher chance to have lower displacement distance, thus the performance is lower.

SOB-R failed to simulate the set-size effect on the intrusion probe. The set-size effect simulated by SOB-R is shown at Figure 9. The simulated intrusion cost decreases monotonically with set size instead of increasing with set size. The reason that SOB-R failed to simulate the set-size effect on the intrusion cost is because the probability of having short displacement distance is higher at lower set size. The amount of interference is determined by the context overlap and the displacement distance. Because the amount of context overlap is constant across set sizes, the displacement distance is the major factor to determine the intrusion cost. For lower set sizes, the maximum displacement distance is limited, thus the intrusion cost is greater comparing to larger set sizes.

3.3 Extralist-feature effect

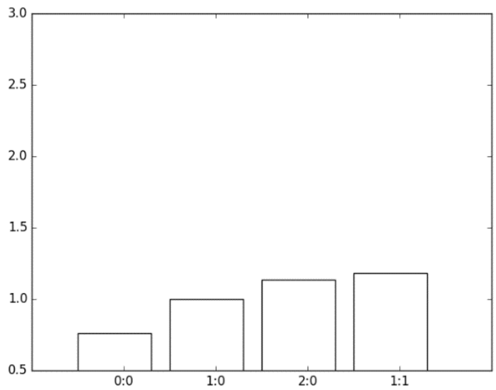

Figure 19. The extralist-feature effect simulated by SOB-R. The extralist-feature effect is simulated with the parameters set of extralist-feature listed in Table 1.

To simulate the extralist-feature effect, the content creation had to be altered in order to fit the stimulus design in the task. The content representation was equally divided into three subsets of nodes, and each subset consisted of 40 units. The three subsets represented the first feature, the second feature, and the conjunction feature. The conjunction feature represents the specific combination of the first and the second features. Each subset was created independently and followed the standard method of content creation. The first and the second subsets reflected the state of the first and second features. For example, item *Aa* and *Ab* had the same first subset of values and different second subsets. The conjunction feature, however, was unique to specific first and second feature combination. Unless items had the same first and second feature, the conjunction feature was always different. The conjunction feature is required for successfully rejecting the 1:1 probe. Both the 1:1 probe and the positive probe would have a match from the first feature and a match from the second feature, and the difference between the 1:1 probe and the positive probe is that the two features in the positive probe came from the same item but the features in the 1:1 probe did not. Without the conjunction feature, the combination of the features in the items is not taking into account when comparing the probe to the retrieved memory, which results in the model fail to distinguish between the 1:1 probe and the positive probe.

The parameters used in the simulation are listed in the Table 1. The simulation consisted of 1000 iterations. The encoding sequence was randomized for each iteration, and all the possible probes were tested during the recognition phase. The performance of each probe type is shown in Figure 19. The 0:0 probe was the easiest probe to be rejected and followed by the 1:0 probe. However, the 1:1 probe and 2:0 probe were simulated with the same performance and were the most difficult probes to be rejected.

SOB-R failed to simulate the performance observed from the 2:0 probe. To be more precise, SOB-R failed to reproduce the difference between the 2:0 probe and 1:1 probe. Both types of probe were equally difficult to be rejected. This is because the retrieved memory is the weighted sum of all the memory items. To simplify the situation by assuming the retrieved context has been fully deblurred and the amount of interference from the shadow is at minimum, the retrieved memory, v_p, can be formalized as:

$$v_p' = C_p p_p, \tag{21}$$

where the context p_p is the deblurred context and suffers no interference from the shadow. Since all the binding are superimposed in the memory network C_p. Equation 21 can be expanded to

$$\begin{aligned} v_p' &= \sum_i^n \eta_i \cdot v_i p_i^T \cdot p_p \\ &= \sum_i^n \eta_i p_i^T p_p \cdot v_i, \end{aligned} \tag{22}$$

where n is the set size of the current trial. The retrieved memory v_p' is the weighted sum of all the memory items, and the weighting is determined by the encoding strength η_i and the dot product of $p_i^T p_p$. Because the retrieved memory is the weighted sum of all the memory items, every matching feature counts toward the evidence of accepting the probe when comparing the probe and the retrieved memory. The 1:1 probe has two matching features, which is the same as the 2:0 probe, thus the performance of the 1:1 probe is at the same level as the 2:0 probe. Thus, SOB-R does not predict the extralist-feature effect, according to which the number of extra-list features determines the performance of rejecting the probe.

3.4 The expectation effect on the serial-position curve

Previous research found that participants' expectation about the upcoming task affects the observed serial-position effect. In the Duncan & Murdock (2000) study, participants were asked to do a recognition task or a serial-recall task. The required task changed from trial to trial, and a cue was presented to instruct the task of the current trial. The cue was either presented before the encoding phase or after the encoding phase. When the task cue was presented before the encoding phase, participants knew the upcoming task before encoding items, thus participants could adjust their encoding strategy accordingly. In the case of the task cue was presented after the encoding phase, participants did not know the upcoming task, thus participants can only encode the items in the way which can accomplish both tasks, which in turn affects the performance of both tasks. The result showed that if participants are able to expect the upcoming task before encoding, the performance of recognition task showed the typical serial-position effect: performance was better at the beginning and the end of the serial positions. However, if participants did not know the upcoming task, a flat serial-position effect in the recognition task was observed, the performance was no different across serial positions, as shown in Figure 20 (left). The performance of the serial-recall task did not differ regardless the of type of task cues. However, a follow up study (Murdock & Duncan, 2003) showed that if participants expected recognition task during encoding but were asked to perform serial-recall task, participants' performance in the serial recall task suffered drastically, as shown in Figure 21 (left). The evidence seems to indicate that the participants can perform recognition task while serial-recall task was anticipated, but not vice versa.

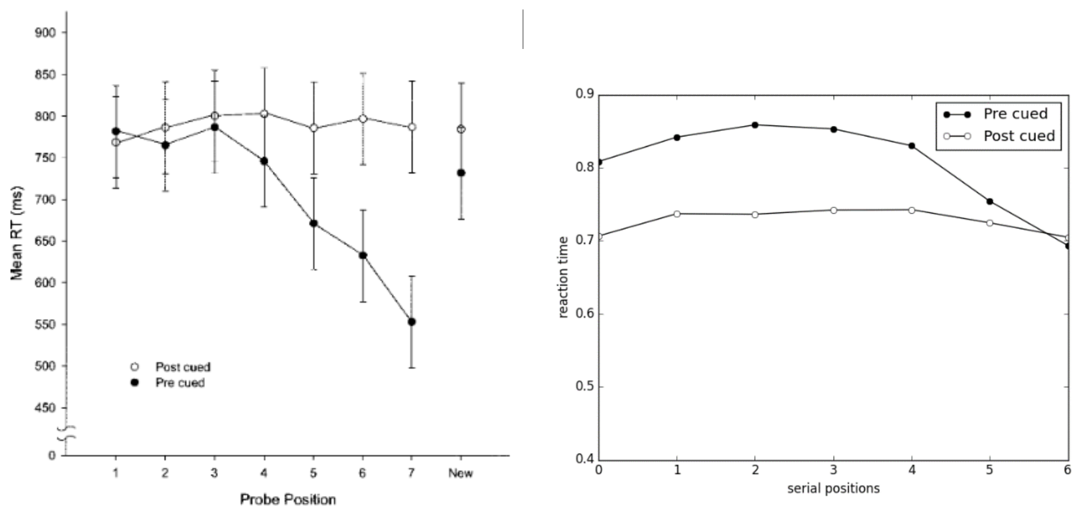

Figure 20. The left figure shows the observed serial-position effect for recognition task when participants expecting a serial-recall and a recognition task. The right figure shows the simulated serial-position effect. The left figure is from Figure 1 of Duncan & Murdock (2000). The simulated serial-position effect is simulated

with the parameters set of Duncan & Murdock's finding listed in Table 1.

Duncan & Murdock's finding is important for SOB-R because one core assumption in SOB-R is that the memory representation is the same for both serial-recall task and recognition task. However, Duncan & Murdock's finding demonstrated that participants' expectation affects the method of encoding, which results in different performance. However, different methods of encoding do not necessarily result in qualitatively different memory representation, but only quantitatively different ones. The memory representation in serial-recall task and recognition task can still both consist of bindings between content and context, but the bindings are formed with different parameters.

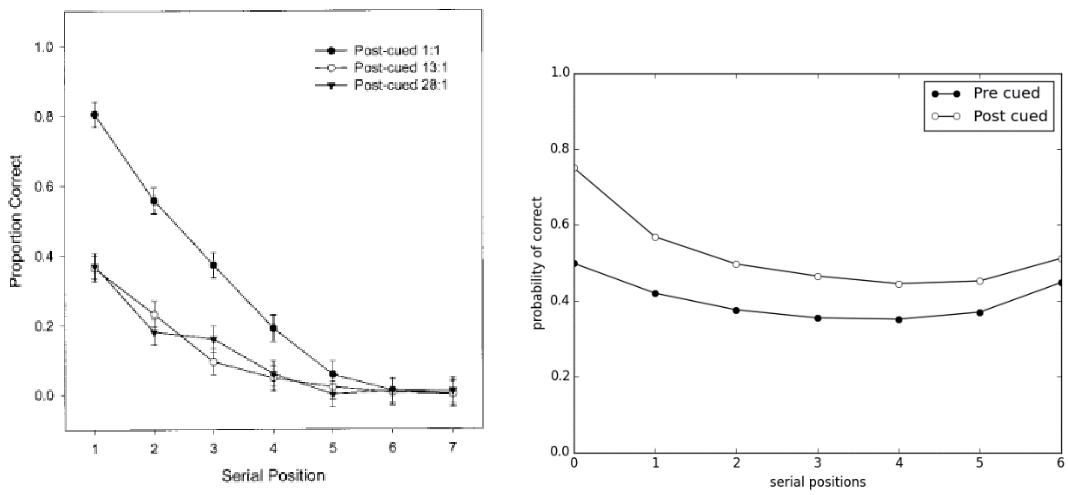

Figure 21. The left figure shows the observed serial-position effect for serial-recall task when participants expecting a serial-recall and a recognition task. The right figure shows the simulated serial-position effect. The left figure is from Figure 3 of Murdock & Duncan, (2003). The simulated serial-position effect is simulated with the parameters set of Duncan & Murdock's finding listed in Table 1.

SOB-R assumes the the amount of interference from the shadow of previously used context changes based on participants' expectation about the upcoming task. If participants expect a serial-recall task, which requires more precise context information, participants have to reduce the influence from the shadow. By reducing the amount of interference from the shadow, each content is bound to its corresponding context with reduced mis-binding. Thus, when activating context to retrieve the bound content, the retrieved memory suffers less interference from mis-binding, which results in better performance. In contract to the serial-recall task; the recognition task, global recognition task in this case, does not require specific context information. The only context information required in the recognition task is whether the probe was presented in the trial or not. The serial position of the probe is not needed. Thus, the mis-binding during the encoding phase does not affect the performance as much.

Thus, SOB-R simulates the expectation effect by modifying the amount left over from the previous context, res_p. The amount of leftover is higher if participants expect a recognition trial. The shadow has less effect if participants expect a serial-recall trial. The simulated serial position effect of the recognition task is shown in Figure 20 (right). With larger res_p, the simulated serial-position effect for the recognition task matches typical finding in the recognition task. With smaller res_p, however, the simulated serial-position effect is flat, i.e., the performance is constant across serial positions. SOB-R simulates the recency gradient in the serial-position effect through the interference of the shadow. With minimum influence from the shadow, the simulated recency gradient is also reduced. That said, SOB-R still simulates serial-position effect with the primacy gradient through the primacy gradient in the encoding strength. The simulated primacy gradient in serial-position effect is relatively small, thus the simulated serial-position effect is still flat across all serial positions.

The simulated serial-position effect for the serial-recall task is shown in Figure 21 (right). As expected, the performance of serial-recall task with small res_p shows the typical serial-position effect in the serial-recall task. However, with larger res_p, the performance of serial-recall task drops drastically. This is because of the increased amount of mis-bindings during the encoding process, and the increased interference from the shadow during the retrieval process. During the encoding process, the content is not only bound to the intended context but also the previously encountered contexts. During the retrieval process, the context used to retrieve bound content contains both the intended context and the shadow. The shadow reduces the precision of binding in both encoding and retrieval process, resulting in poorer performance.

In Duncan & Murdock's study, participants' performance in the recognition task was better if participants expected a recognition trial comparing to the performance if participants expected a serial-recall trial. However, SOB-R failed to simulate this finding. The simulated performance when participants expected a serial-recall task is better compared to the performance when participants expected a recognition trial. This is because the amount of interference from the shadow is larger when expecting a recognition trial. When retrieving the expectation of the probe, non-target items are also retrieved because of the interference from the shadow. Thus, the retrieved memory has more interference from non-target items with stronger interference from the shadow, which results in worse performance in the recognition task.

3.4 Continuous stimulus

Most studies of short-term recognition used discrete items as stimulus, e.g. characters, words, or pictures (Lange, Cerella, & Verhaeghen, 2011; Lange, Verhaeghen, & Cerella, 2010; Schwartz, Howard, Jing, & Kahana, 2005). Some studies used items with continuous features, e.g. orientation, color, or spatial frequency (Kahana & Caplan, 2002; Kahana & Sekuler, 2002; Kahana et al., 2007). Those studies found that the probability of accepting the probe as an old item is determined by the similarity between the probe and the memory items. NEMO and EBRW can easily simulate those findings through choosing the right locations of items in the multi-dimensional feature-space. Because feature dimensions are assumed as continuous, NEMO and EBRW can simulate the continuous change in the feature dimensions by simply changing the feature values of dimensions. Moreover, because the similarity between items is measured by the distance between items in the multi-dimensional feature-space, changing the feature values also results in continuously changing the similarity between items.

Although SOB-R was originally designed for simulating tasks with discrete stimuli, SOB-R can also simulate the recognition task with continuous stimuli by manipulating the similarity between content representations during the content creation. In SOB-R, each content is represented as a vector of values. As demonstrated in the simulation of the extralist-feature effect, a feature can be represented by a subset of units. By manipulating the similarity between subsets, SOB-R can manipulate the similarity between items in the corresponding feature, which enables SOB-R to have continuous features.

To construct the item representation, I assumed that an item consists of two independent features without conjunction features. An item was represented by 120 units, and each feature consisted of 60 units. The 60 units of the feature were used to represent a single value of that feature. The similarity between two features values should be reflected in their corresponding vectors of unit, i.e. the cosine between vectors. To create the vector which can reflect the similarity between feature values, the vector of units was distributed as normal distribution centered at the value of the feature, the individual unit j of feature i was determined via:

$$v_{i,j} = \frac{1}{s_v}\phi\left(\frac{j - x_i}{s_v}\right), \tag{23}$$

where the ϕ is the standard normal distribution centered at the value of feature i, x_i, with a standard deviation of s_v, as shown on the left side of Figure 22. The similarity between two vectors was a function of the difference between the feature values of the corresponding vectors. The similarity between vectors reduced when the difference between feature values increased, while the amount of decreasing in similarity was modulated by s_v, which represented the sensitivity of changes in the feature dimension, as shown in Figure 22 (right).

Smaller s_v generateed a more peaked distribution in the vector of nodes, thus a small amount of shift resulted in huge change in the vector, which in turn resulted in huge similarity drops with small changes. Larger s_v generated a much flattened distribution, which resulted in smaller similarity drop under the same amount of shift in the feature value, as shown in Figure 22 (right).

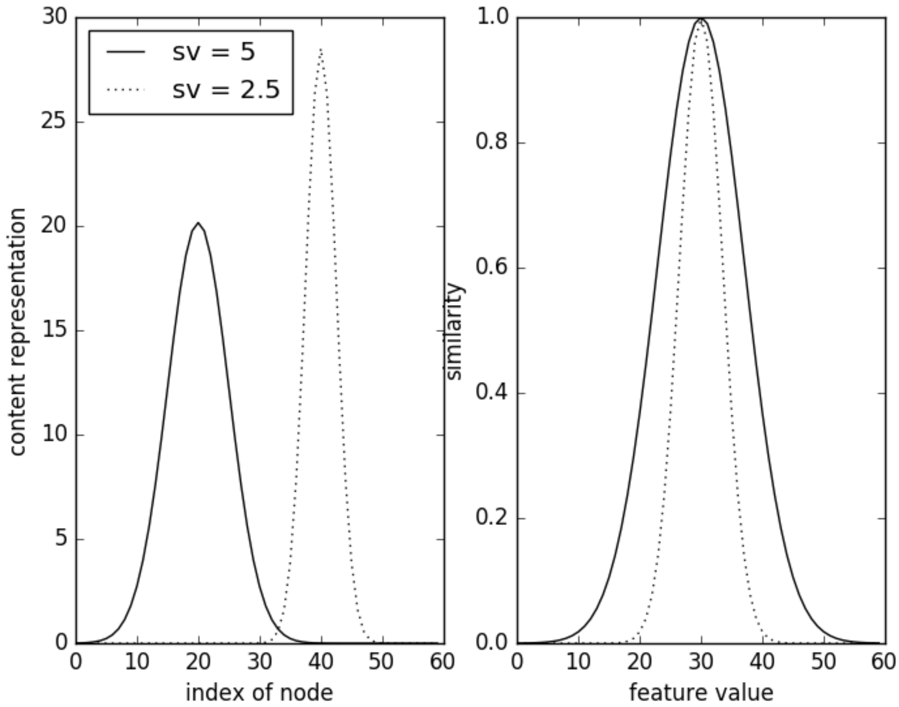

Figure 22. The left figure shows the values of nodes under different s_v, and the right figure shows the similarity between an item with feature value 30 and an item with different feature values under different s_v.

In the following simulation, I demonstrated that SOB-R can also apply to continuous stimuli. The simulation consisted of 100 iterations. The parameters used in the simulation were listed in the Table 1. In each iteration of simulation, four items were encoded in the memory in random order. Each item consisted of two independent features. Each feature had one value which ranges from 1 to 60. The value of the feature represents the state of feature on the corresponding dimension, e.g. 3.5 Hz in spatial frequency. The four items encoded in the memory had feature values of (20, 20), (20, 40), (40, 20), and (40, 40), where the first and the second numbers represented the values of first and second features, respectively. After encoded four items, 60×60 probes were tested, where each probe occupies one possible combination of values of the two features.

The simulation result is shown in Figure 23. The smaller similarity between the probe and the memoranda resulted in lower probability of accepting the probe as an old item. This pattern is similar to the findings in the short-term recognition studies with continuous material. The probability of accepting the probe as an old item decreases when the similarity between the probe and the memory items decreases, as shown in Figure 23.

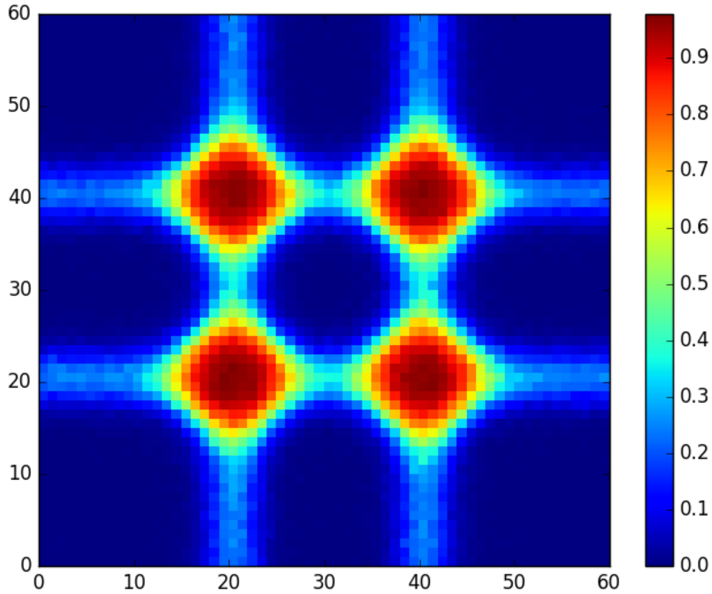

Figure 23. The simulation result of continuous stimulus simulation. The x axis and the y axis represent the feature values of the first and the second dimension. The color represents the probability of accepting the probe. The continuous stimulus simulation is simulated with the parameters set of continuous stimulus in Table 1.

General Discussion

With minimal modifications from SOB-CS, SOB-R is able to simulate a wide range of phenomena in short-term recognition tasks while retaining the ability of simulating the serial-recall task. SOB-R inherits most of the assumptions in SOB-CS with two modifications. The first modification is that SOB-R assumes that energy, i.e., the comparison between the retrieved memory of the outcome and the actual outcome, is calculated by the normalized dot product, i.e. cosine. The second modification is that SOB-R assumes that the usage of context is imperfect due to the interference by the shadow of previously used contexts. With both modifications, SOB-R still retains the ability to simulate the serial-recall task. At the same time, SOB-R can simulate many findings from the recognition tasks, including serial-position effects, speed-accuracy trade-off functions, and the recognition performance with continuous stimuli.

4.1 Failed simulations and ways to fix them

Despite the fact that SOB-R is able to simulate many phenomena in recognition tasks, there are a few effects which SOB-R cannot simulate correctly. SOB-R failed to simulate the linear increasing of reaction time in Sternberg's task. Instead of a linear increase of reaction time, SOB-R predicts an increasing of reaction time with decelerating speed. Also, SOB-R failed to simulate the set-size effect on intrusion costs in local recognition task. SOB-R predicts that the intrusion costs decreases with set size, while research has shown that intrusion costs increase monotonically with set size (Donkin et al., 2014). SOB-R failed to simulate the extralist-feature effect. Lastly, SOB-R failed to simulate the relative overall performance level of the recognition when expecting recognition or recall in the expectation effect on the serial-position curve. However, those simulations were done with many constraints imposed in order to demonstrate the predictions from the core of the model. By lifting a few of the constraints, SOB-R is capable to simulate some of these phenomena.

4.1.1 Linear increasing of the reaction time

SOB-R failed to simulate the linear increase of reaction time in Sternberg's task because the amount of crosstalk, the main source of set-size effect, increases exponentially. The evidence for accepting or rejecting the probe is the result of comparing the probe and the expectation of the probe. The expectation of the probe comes from activating the context to retrieve the bounded content. Because the context overlapping with each other, as defined in Equation 1, the amount of interference does not increase linearly, which results in a non-linear decrease of evidence across set sizes. One constraint on simulating the set-size effect in SOB-R is that all the parameters in the interface model remain constant across set sizes. Only the drift rate varies according to the evidence predicted by the model. Thus the simulated set-size effect reflects the simulation from the core model instead of the interface model.

However, by lifting the constraint on the interface model and assuming that the decision boundary, a, increases with set size, SOB-R is able to simulate the linear increase of reaction time. Previous research has shown that the set-size effect is not only reflected in the drift rate but also the boundary separation (Ratcliff, 1978). Participants adjust the level of caution based on the set size of the trial, which is reflected in the estimated boundary separation. However, because SOB-R can only simulate different drift rates for different set sizes, the boundary separation remains the same across set size. By releasing the constraint of using a constant boundary separation, the boundary separation is assumed as linearly increasing across set size via

$$a_n = a_0 + n \times a_{caution}, \tag{24}$$

where the a_n is the boundary separation at set size n, and the $a_{caution}$ is the amount of caution increases at each set size. After allowing the boundary separation to increase with set size, the simulated set-size effect is shown in left side of Figure 24. The reaction time increases linearly across the set sizes, because the boundary separation is larger at larger set size, which increases the time required to accumulate enough evidence to make response. The increase of boundary separation also affects the simulated accuracy. The accuracy at larger set sizes increases comparing to the restrained simulation.

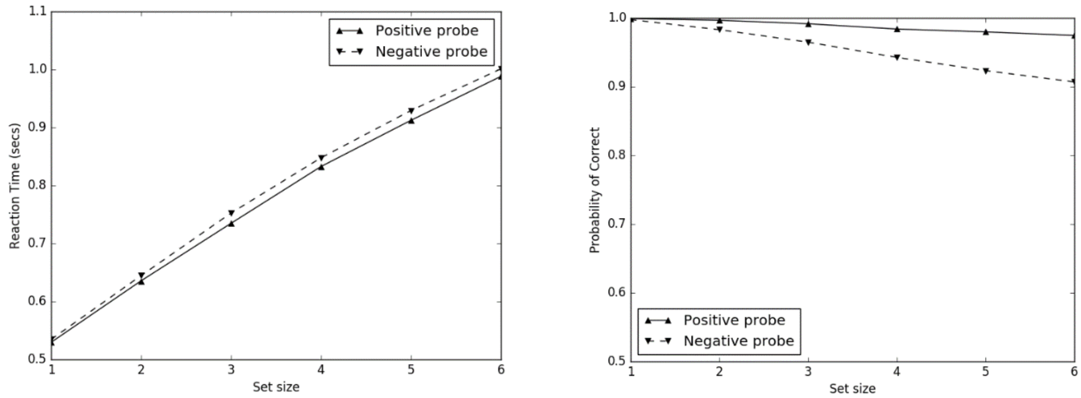

Figure 24. The set-size effect simulated with variant a in the interface. $a_0$is set to 0.166, and $a_{coution}$ is set to 0.014. The left figure shows the set-size effect on the reaction time, and the right figure shows the set-size effect on the accuracy.

4.1.2 Set-size effect on intrusion cost

SOB-R also failed to simulate the set-size effect on intrusion costs. SOB-R predicts that intrusion costs decrease with set size. However, previous research (Donkin et al., 2014) has shown the opposite pattern, that intrusion cost increases monotonically with set size. The reason that SOB-R makes the opposite prediction is that the strongest intrusion cost occurs when the intrusion probe comes from the neighboring position. The context overlaps with the neighboring context the most, as defined in Equation 1. When using context to retrieve the bound content, the content at other positions are also retrieved. The amount of interference generated from the content is determined by the distance between the activated context and the context of the content. Thus the amount of intrusion cost increases when the distance between the position of origin and the test position decreases, as shown in Figure 16. With smaller set size, most intrusion probes are from close neighbors, which results in higher intrusion costs. The intrusion probes have higher probability to be from longer distance in larger set size, which results in lower intrusion costs. Because SOB-R assumes the context overlapping is constant across set sizes, the predicted intrusion cost decreases with set size.

Though it is reasonable to assume that the context overlap is constant across set sizes in Sternberg's task, the same might not be true in the local recognition task. In Sternberg's task, the participant does not know the set size of the current trial until the end of the learning phase. The memoranda are presented sequentially until an end-of-list signal is presented, which indicates the end of learning phase. Thus the participant is not able to know the set size of the current trial while encoding the memoranda. In most of the local recognition tasks, participant knows the set size of the current trial before the learning phase. At the beginning of the trial, a few empty frames are presented on the screen, which indicate the locations where the items will be presented. Thus the number of empty frames also foretells the set size of the trial. When the participant does not know the set size of the current trial, participant cannot optimize the usage of context used for binding the items according to the set size. However, when the participant knows the set size of the current trial, the participant can adjust the usage of context to optimize the performance for the set size. For example, in the case of optimizing the usage of context in the trial of set size 2, participant can bind the first item to the first serial position and then bind the second item to the sixth serial position, which minimizes the overlap between the contexts of first and second items and results in better performance. The same strategy cannot be applied to the situation in which the participant does not know the set size in advance.

Instead of assuming that the context overlap is constant across set sizes, the following simulation assumes that the context overlap varies at different set sizes. The context overlap is defined as

$$s_p(n) = s_p(nmax)^{\frac{nmax-1}{n-1}}, \tag{25}$$

where the $s_p(n)$ is the context overlap at set size n. $s_p(n)$ is determined by the context overlap at the the maximum set size, $nmax$. The assumption is that the minimum context overlap would be $s_p(nmax)^{nmax-1}$, and the participant can optimize the usage of context based on the current set size of the trial. For example, assuming the participant can optimize the usage of context overlapping up to set size 7 and the contexts used at set size 7 are p_1, p_2, p_3, … p_7. When the participant encountered set size 4, the context p_1, p_3, p_5, and p_7 are used to bind the items from the 1st, 2nd, 3rd, and 4th serial positions, respectively. The overlap between neighboring positions in set size 4, $s_p(4)$, is the same as the overlapping between serial position 1 and 3, which is $s_p(7)^2$.

The simulated set size effect is shown in Figure 25. The intrusion cost increases from set size 1 to set size 4 and flattens after set size 4. The same pattern is also found in previous studies (Donkin et al., 2014). The increase from set size 1 to set size 4 is because the context overlap between neighboring positions increases from set size 1 to set size 4. After set size 4, although the amount of context overlap still increases with set size, the probability of having intrusion probes from far distance also increases. Thus the effect of increasing the context overlap and the effect of increasing the probability of having larger distances cancel each other, which results in a flat set-size effect on intrusion costs.

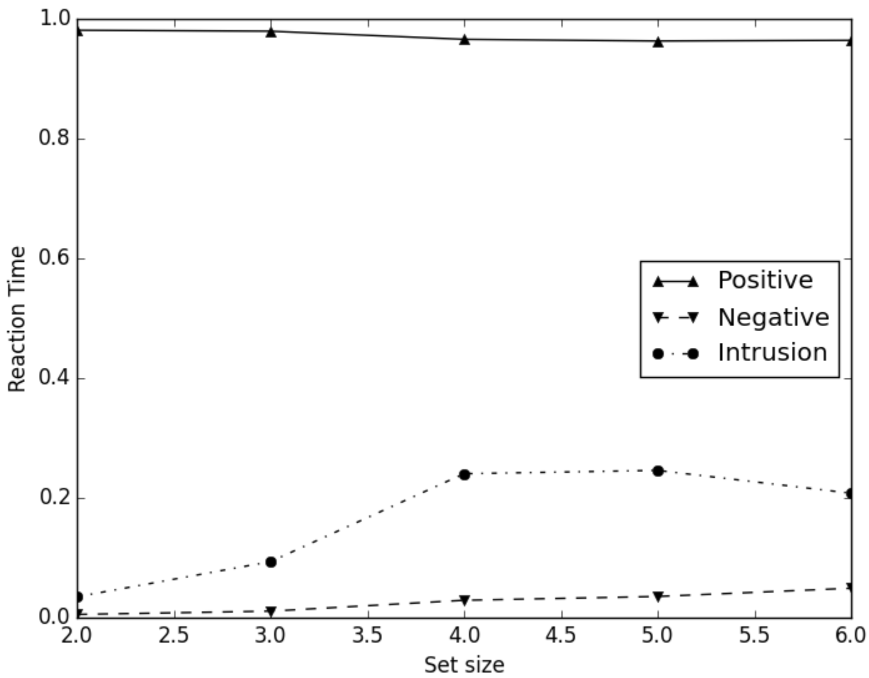

Figure 25. The set-size effect in local recognition task simulated with variant context overlapping. The $nmax$ is 7, and the $s_p(7)$ is 0.5.

4.1.3 Extralist-feature effect

SOB-R also failed to simulate the extralist-feature effect. SOB-R predicted that the 2:0 probe (i.e. the new probe that has one matching feature that appeared twice in the memoranda, and one extralist feature) has the same level of performance as the 1:1 probe (i.e. the new probe that has two matching features that both appeared once in the memoranda, and no extralist feature). However, the studies showed that the 2:0 probe is easier to be rejected compared to the 1:1 probe, and the performance of 2:0 probe is the same as the 1:0 probe (i.e. the new probe that has one matching feature that appeared once in the memoranda).

SOB-R failed to simulate the extralist-feature effect because SOB-R performs similar to a summed-similarity model. According to Equation 22, every appearance of a matched feature in the memoranda counts toward the evidence of accepting the probe. The 2:0 probe has one matching feature that appeared twice in the memoranda, thus the performance of the 2:0 probe is determined by the two feature appearances. The 1:1 probe also has two feature appearances, which is the result of two matching features that both appeared once, thus the predicted performance of the 2:0 probe is the same as the 1:1 probe.

SOB-R is not the only model that cannot simulate the extralist-feature effect. EBRW also failed to simulate the extralist-feature effect. Because EBRW is also a summed-similarity model, EBRW fails for the same reason as SOB-R to simulate the extralist-feature effect. Although NEMO did not attempt to simulate the extralist-effect, because of the summed-similarity nature of NEMO, it is unlikely that NEMO is able to simulate the extralist-feature effect.

IRM is the only model that is capable to simulate the extralist-feature effect. IRM simulates the extralist-feature effect because the rejection threshold is not counting the number of matching features but counting the mismatching features. The 2:0 probe has only one matching feature that appeared twice, which leaves one mismatching feature (or two mismatching features if the conjunction feature is taken into account). The 1:1 probe has two matching features, thus zero mismatching feature (or one mismatching features if the conjunction feature is included). This results in the 1:1 probe being the most difficult probe to be rejected. The 1:0 probe has one matching feature and one mismatching feature (or two mismatching features with the conjunction features), thus the 1:0 probe leads to the same performance as 2:0 probe.

4.1.4 The expectation effect on the serial-position curve

Although SOB-R was able to simulate the different serial-position effects based on participants' expectation, SOB-R failed to simulate the finding that performance in the recognition task is better when participants expected a recognition trial than the performance when participants expected a serial-recall task. SOB-R simulates the flat serial-position effect by reducing the amount of interference from the shadow of the previous used context. SOB-R assumes that the interference from the shadow is smaller if participants expect to encounter a serial-recall trial. Because the shadow is the major source of serial-position effect, by reducing the interference from the shadow, the predicted performance becomes constant across the serial positions. However, reducing the interference from the shadow also reduces the interference from the non-target items, because the binding is more accurate during the encoding and less non-target items are retrieved during the recognition. Reducing the interference from the shadow results in increasing performance in the recognition task. On the other hand, when participants expect a recognition trial, the amount of interference from the shadow is less, which results in reducing performance in the recognition task.

Because SOB-R measures the similarity between probe and expectation by taking the cosine of both vectors, the amount of interference directly affects the similarity. However, despite the increasing amount of interference in the expectation, the similarity between probe and expectation still leads to the right response about accepting or rejecting the probe. Because the interference in the expectation consists of non-target items, the average of the non-target items becomes the prototype item. Since the threshold of accepting or rejecting the probe is set to the average similarity between items and the prototype item, increasing the interference will push the similarity between the expectation and the probe toward the threshold but not across the threshold, thus the similarity between the expectation and the probe is always valid for accepting and rejecting the probe regardless of the amount of interference.

Assuming participants are aware of the increasing amount of interference in the expectation, participants can compensate it by reducing the decision boundary, a, in the interface model. Because the amount of interference simply reduces the amount of valid information in the evidence, thus, by shrinking the decision boundary, the decision can be made faster without sacrificing too much accuracy. When participants expect a recognition trial, the simulation result is shown at Figure 26. When participants expect the upcoming trial is a recognition trial, the simulated performance is better than the simulated performance when participants expect a serial-recall trial.

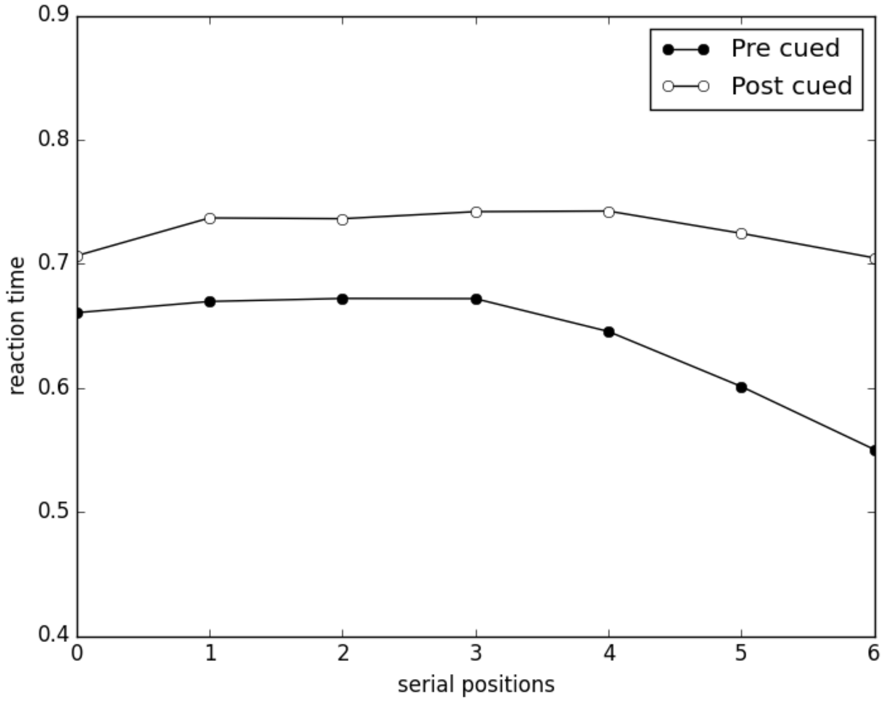

Figure 26. The simulated serial-position effect with smaller $a = 0.16$ when participants expect a recognition trial. $a = 0.2$ if participants expect a serial-recall trial.

Another possible explanation of the faster responses of the recognition task when participants expected a recognition trial is the omission of the task-switching cost. If participants expected a serial-recall trial but encountered a recognition trial, participants had to switch their expected action of response from serial recall to recognition. For serial-recall task, participants have to recall the memory items, thus the response is typing all the memory items on the keyboard. For the recognition task, participants simply have to judge whether the probe is in the list or not, and the response is made by pressing one of the two buttons which indicate “old” and “new”. Because two tasks have different response schemes, if participants expected one response scheme but encountered another task, participants had to switch to the required response scheme, which takes extra time for switching. The extra time is the task-switching cost. If participants expected a recognition trial and encountered a recognition trial, participants did not suffer from the task-switching cost because the expected and encountered tasks are consistent. However, if participant expected a serial-recall trial and encountered a recognition trial, participants had to switch to another response scheme and suffered from the task-switching cost. Thus, the average reaction time in the recognition task is longer if participants expected a serial-recall trial.

To simulate the task-switching cost when participants expected a serial-recall task, the task-switching cost can be added on the non-decision time, T_{er}. By adding the task-switch time to non-decision time, the reaction time on the recognition task when participants expected a serial-recall trial increases. Because the task-switching cost is applied constantly across serial positions, the pattern of the serial-position effect does not change. The simulation still shows the flat serial-position effect when participants expected a serial-recall trial. The simulation is shown in Figure 27.

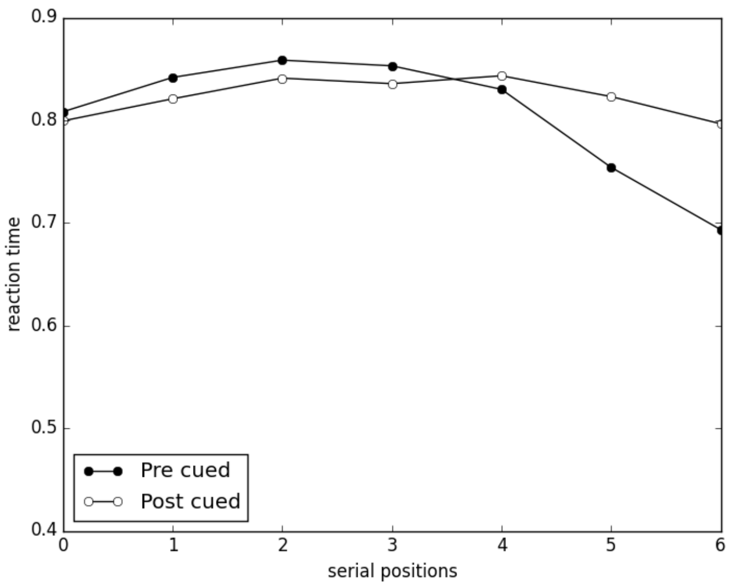

Figure 27. The simulated serial-position effect with larger $Ter = 0.4$ if participants expect a serial-recall trial. $Ter = 0.3$ if participants expect a recognition trial.

4.2 Comparing SOB-R to the other recognition models

One of the intentions for developing SOB-R is to fill in the missing mechanisms in the previous short-term recognition models. All the recognition models did not explain the difference in the activation strength between serial positions. NEMO and EBRW also employed the different decision criterion at different set sizes. SOB-R, on the other hand, was able to simulate a wide range of phenomena without using free parameters to simulate the serial-position effect and the set-size effect. SOB-R is also able to simulate the local recognition task, which was not simulated by the other recognition models.

4.2.1 Activation strength between serial positions

To simulate the serial-position effect, NEMO, EBRW, and IRM rely on different activation strengths of items at different serial positions. However, the reason behind different activation strengths was left unexplained. NEMO, EBRW, and IRM simply assumed the activation strengths are free parameters. The items in SOB-R are also encoded with different encoding strength based on their serial positions. The different encoding strength is the result of energy-gated encoding. Because the energy (novelty) of the to-be-encoded item gradually decreases when more items were encoded in the memory, the encoding strength decreases along with the serial positions.

The energy-gated encoding is only one factor which affects the serial-position effect. The energy-gated encoding results in primacy gradient in the activation strength, which results in the primacy gradient of the serial-position effect. The recency gradient in the serial-position effect is simulated though the shadow of the previous used contexts.

4.2.2 Constant threshold between set sizes

The NEMO and EBRW models assume the threshold for determining the acceptance or rejection of the probe varies depending on the inter-item similarity of the current trial. Instead of relying on different thresholds on different set sizes, SOB-R assumes the threshold is constant across the set sizes. The threshold is set as the average similarity between the items and the prototype of items. While retrieving the content from the memory by activation a context, the overlap between contexts results in retrieving not only the item bound to the context but also the items from the other serial positions. When the set size increases, more items are encoded in the memory, thus the retrieved memory will be more similar to the prototype of the items, which is more difficult to be correctly accepted or rejected.

4.2.3 Simulating the local recognition task

SOB-R is able to simulate the serial-position effect in the local recognition task. With a small modification in the amount of context overlap, SOB-R is also able to simulate the set-size effect of the intrusion cost. Local recognition task requires not only the item information, i.e. whether the probe is in the memory list or not, but also the context information, i.e. whether the probe is presented in the correct context. NEMO, EBRW, and IRM can incorporate the context information as part of the item representation if the task requires. Since NEMO and EBRW assume that item memory exists in a multidimensional feature space, one or more of the dimensions can represent the context information. Because the activation of a probe is the sum of the similarity between probe and items across all the dimensions, the activation of the probe also involves the dimensions of context information, if the weighting of the dimensions of context information are larger than zero. SIMPLE, a memory model which also assumes item memory exists in multiple-dimensional feature-space, uses one dimension of context information, i.e. time, as a main source of interference. The same mechanisms can also be implemented in NEMO and EBRW. IRM can treat the context information as features and encode the context information along with the other features in the item. The encoded context information will then influence the recognition process and affect the decision. The amount of influence from context information is determined by the number of elements used to represent the context information.

Although NEMO and EBRW can incorporate the context information, the serial-position effect simulated by NEMO and EBRW could be inconsistent with the previous finding. Both Nemo and EBRW are essentially the single-process model in Oberauer (2008). The context information and the content information cannot be separated during the recognition process. For intrusion probes, the serial-position effect simulated by the single-process model shows the opposite pattern from the observed result when aggregating the result according to the position of origin.

4.3 Conclusion and Outlook

SOB-R inherited most of the assumptions in SOB-CS and assumes that memory representations in the serial-recall task and the recognition task are not qualitatively different. The only difference is the amount of influence from the shadow of previous used contexts. The shadow has stronger influence in the recognition task and weaker influence in the serial-recall task. Without alternating too many assumptions in the SOB-CS, SOB-R is able retain the ability of simulating the serial-recall task and many phenomena in the recognition tasks. To my knowledge, SOB-R is the only short-term memory model which can simulate both serial-recall task and the recognition task.

Compared to the other recognition models, SOB-R is able to simulate most of the phenomena simulated by the other recognition models. The difference of the activation strength between serial positions is also explained by SOB-R, which was not explained by the other models. SOB-R also uses a constant threshold for accepting or rejecting the probe. Although SOB-R failed to simulate the extralist-feature effect, SOB-R is able to simulate the result of the local recognition task, which was not simulated by the other recognition models.

One of the new introduced mechanism is the shadow of the previous used context. The shadow contributes to the recency gradient in the serial-position effect, and the amount of influence from the shadow is different between the serial-recall task and the recognition task. SOB-R assumes that participants' expectation about the incoming task would affect the amount of influence from the shadow, as simulated in the expectation effect on the serial-position curve.

The shadow of the previous used context is not only used to explain previous findings. The shadow also provides unique predictions about the serial-position effect. The shadow predicts that the serial-position effect in the order reconstruction task shows stronger primacy gradient and weaker recency gradient, and the serial-position effect in the probed recall task shows stronger recency gradient and weaker primacy gradient. Although previous findings (Kahana & Caplan, 2002; Neath, 1997) were consistent with the prediction from SOB-R, the order-reconstruction task and the probed recall task were tested in different experiments. Few previous studies tested both item-probe order recall and the order-probe item recall in the same experiment. The result from those experiments, however, are not consistent with each other (Detterman, 1977; Jones, 1976; Nairne, Whiteman, & Woessner, 1995).

This is an important test to the shadow mechanism in SOB. Because without the influence from the shadow, the serial-position effect from the order-reconstruction task and the probed recall task should show similar pattern. Without the influence of the shadow, the serial-position effect from both tasks is modulated by the strength of the bindings. Regardless of whether one is recalling the position from the item or recalling the item from the order, because the strength of binding should not differ, the predicted serial-position effect would show a similar pattern. Because previous studies tested the order-reconstruction task and the probed recall task in different experiments, there is no way to ensure that the binding strengths were the same among both tasks. The difference in the serial-position effect could be the result of different binding strength instead of the influence from the shadow.

To ensure the binding strength is the same for both tasks, both tasks can be tested in the same experiment. Since the learning phase of both tasks are the same, participants can go thought the same learning phase without knowing the task of the current trial. After participants learned the memoranda, they encounter either an order-reconstruction task or a probed recall task. Because participants could not expect the incoming task, the binding strength should be constant across the two tasks. If the disparity in the serial-position effect is observed between tasks, it cannot be explained by the difference of binding strength. There must be other mechanism involved, and one of the possible mechanism is the shadow of the previous context.

References

Awh, E., Barton, B., & Vogel, E. K. (2007). Visual Working Memory Represents a Fixed Number of Items Regardless of Complexity. *Psychological Science*, *18*(7), 622–628. http://doi.org/10.1111/j.1467-9280.2007.01949.x

Brainerd, C. J., Reyna, V. F., & Kneer, R. (1995). False-Recognition Reversal: When Similarity is Distinctive. *Journal of Memory and Language*, *34*(2), 157–185. http://doi.org/10.1006/jmla.1995.1008

Brown, G. D. A., Preece, T., & Hulme, C. (2000). Oscillator-based memory for serial order. *Psychological Review*, *107*(1), 127–181. http://doi.org/10.1037/0033-295X.107.1.127

Cleary, A. M., Morris, A. L., & Langley, M. M. (2007). Recognition memory for novel stimuli: The structural regularity hypothesis. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *33*(2), 379–393. http://doi.org/10.1037/0278-7393.33.2.379

Cowan, N., Rouder, J. N., Blume, C. L., & Saults, J. S. (2012). Models of verbal working memory capacity: What does it take to make them work? *Psychological Review*, *119*(3), 480–499. http://doi.org/10.1037/a0027791

Detterman, D. K. (1977). A Comparison of Item and Position Probes in Short-Term Memory. *The American Journal of Psychology*, *90*(1), 45–53. http://doi.org/10.2307/1421639

Donkin, C., & Nosofsky, R. M. (2012). A Power-Law Model of Psychological Memory Strength in Short- and Long-Term Recognition. *Psychological Science*, *23*(6), 625–634. http://doi.org/10.1177/0956797611430961

Donkin, C., Tran, S. C., & Pelley, M. L. (2014). Location-based errors in change detection: A challenge for the slots model of visual working memory. *Memory & Cognition*, 1–11. http://doi.org/10.3758/s13421-014-0487-x

Duncan, M., & Murdock, B. (2000). Recognition and Recall with Precuing and Postcuing. *Journal of Memory and Language*, *42*(3), 301–313. http://doi.org/10.1006/jmla.1999.2690

Farrell, S., & Lelièvre, A. (2012). The dynamics of access to groups in working memory. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *38*(6), 1659–1674. http://doi.org/10.1037/a0028469

Gilchrist, A. L., & Cowan, N. (2014). A two-stage search of visual working memory: investigating speed in the change-detection paradigm. *Attention, Perception, & Psychophysics*, *76*(7), 2031–2050. http://doi.org/10.3758/s13414-014-0704-5

Hay, D. C., Smyth, M. M., Hitch, G. J., & Horton, N. J. (2007). Serial position effects in short-term visual memory: A SIMPLE explanation? *Memory & Cognition*, *35*(1), 176–190. http://doi.org/10.3758/BF03195953

Hebb, D. O. (1949). *The Organization of Behavior: A Neuropsychological Theory*. Psychology Press.

Hintzman, D. L. (1984). MINERVA 2: A simulation model of human memory. *Behavior Research Methods, Instruments, & Computers*, *16*(2), 96–101. http://doi.org/10.3758/BF03202365

Jahnke, J. C., Davis, S. T., & Bower, R. E. (1989). Position and order information in recognition memory. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *15*(5), 859–867. http://doi.org/10.1037/0278-7393.15.5.859

Jones, G. V. (1976). A fragmentation hypothesis of memory: Cued recall of pictures and of sequential position. *Journal of Experimental Psychology: General*, *105*(3), 277–293. http://doi.org/10.1037/0096-3445.105.3.277

Kahana, M. J., & Caplan, J. B. (2002). Associative asymmetry in probed recall of serial lists. *Memory & Cognition*, *30*(6), 841–849. http://doi.org/10.3758/BF03195770

Kahana, M. J., & Sekuler, R. (2002). Recognizing spatial patterns: a noisy exemplar approach. *Vision Research*, *42*(18), 2177–2192. http://doi.org/10.1016/S0042-6989(02)00118-9

Kahana, M. J., Zhou, F., Geller, A. S., & Sekuler, R. (2007). Lure similarity affects visual episodic recognition: Detailed tests of a noisy exemplar model. *Memory & Cognition*, *35*(6), 1222–1232. http://doi.org/10.3758/BF03193596

Lange, E. B., Cerella, J., & Verhaeghen, P. (2011). Ease of access to list items in short-term memory depends on the order of the recognition probes. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *37*(3), 608–620. http://doi.org/10.1037/a0022220

Lange, E. B., Verhaeghen, P., & Cerella, J. (2010). Dual representation of item positions in verbal short-term memory: Evidence for two access modes. *European Journal of Cognitive Psychology*, *22*(3), 463–479. http://doi.org/10.1080/09541440903155658

McElree, B., & Dosher, B. A. (1989). Serial position and set size in short-term memory: The time course of recognition. *Journal of Experimental Psychology: General*, *118*(4), 346–373.

Mewhort, D. J., & Johns, E. E. (2000). The extralist-feature effect: evidence against item matching in short-term recognition memory. *Journal of Experimental Psychology. General*, *129*(2), 262–284.

Mewhort, D. J. K., & Johns, E. E. (2003). Sharpening the echo: An iterative-resonance model for short-term recognition memory. *Memory*, *13*(3–4), 300–307. http://doi.org/10.1080/09658210344000242

Murdock, B., & Duncan, M. (2003). Item and associative recognition with precuing and postcuing. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *29*(2), 211–223. http://doi.org/10.1037/0278-7393.29.2.211

Nairne, J. S., Whiteman, H. L., & Woessner, L. (1995). Symmetrical Cuing Effects for Item and Position Information. *The American Journal of Psychology*, *108*(3), 345–358. http://doi.org/10.2307/1422894

Neath, I. (1997). Modality, concreteness, and set-size effects in a free reconstruction of order task. *Memory & Cognition*, *25*(2), 256–263. http://doi.org/10.3758/BF03201116

Nosofsky, R. M., Cox, G. E., Cao, R., & Shiffrin, R. M. (2014). An exemplar-familiarity model predicts short-term and long-term probe recognition across diverse forms of memory search. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, No Pagination Specified. http://doi.org/10.1037/xlm0000015

Nosofsky, R. M., Little, D. R., Donkin, C., & Fific, M. (2011). Short-term memory scanning viewed as exemplar-based categorization. *Psychological Review*, *118*(2), 280–315. http://doi.org/10.1037/a0022494

Nosofsky, R. M., & Palmeri, T. J. (1997). An exemplar-based random walk model of speeded classification. *Psychological Review*, *104*(2), 266–300. http://doi.org/10.1037/0033-295X.104.2.266

Oberauer, K. (2003). Understanding serial position curves in short-term recognition and recall. *Journal of Memory and Language*, *49*(4), 469–483. http://doi.org/10.1016/S0749-596X(03)00080-9

Oberauer, K. (2008). How to say no: Single- and dual-process theories of short-term recognition tested on negative probes. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *34*(3), 439–459. http://doi.org/10.1037/0278-7393.34.3.439

Oberauer, K., Lewandowsky, S., Farrell, S., Jarrold, C., & Greaves, M. (2012). Modeling working memory: An interference model of complex span. *Psychonomic Bulletin & Review*, *19*(5), 779–819. http://doi.org/10.3758/s13423-012-0272-4

Ratcliff, R. (1978). A theory of memory retrieval. *Psychological Review*, *85*(2), 59–108. http://doi.org/10.1037/0033-295X.85.2.59

Schwartz, G., Howard, M. W., Jing, B., & Kahana, M. J. (2005). Shadows of the Past. *Psychological Science (Wiley-Blackwell)*, *16*(11), 898–904. http://doi.org/10.1111/j.1467-9280.2005.01634.x

Shannon, C. E. (2001). A Mathematical Theory of Communication. *SIGMOBILE Mob. Comput. Commun. Rev.*, *5*(1), 3–55. http://doi.org/10.1145/584091.584093

Sternberg, S. (1969). Memory-scanning: Mental Processes Revealed by Reaction-time Experiments. *American Scientist*, *57*(4), 421–457. http://doi.org/10.2307/27828738

Wagenmakers, E.-J., van der Maas, H. L. J., Dolan, C. V., & Grasman, R. P. P. P. (2008). EZ does it! Extensions of the EZ-diffusion model. *Psychonomic Bulletin & Review*, *15*(6), 1229–1235. http://doi.org/10.3758/PBR.15.6.1229

Wixted, J. T. (2007). Dual-process theory and signal-detection theory of recognition memory. *Psychological Review*, *114*(1), 152–176. http://doi.org/10.1037/0033-295X.114.1.152

Yonelinas, A. P. (2002). The Nature of Recollection and Familiarity: A Review of 30 Years of Research. *Journal of Memory and Language*, *46*(3), 441–517. http://doi.org/10.1006/jmla.2002.2864

Acknowledgement

I have no idea how to properly write an acknowledgement. I mean, seriously. I think I'm supposed to thank all the people to help me along the 5 years journey. There are so many people who helped me, although I can only remember 4 of them, which is a shame, but that's what you get when you believe in slot model. I tried to chunk everyone into four slots. Of course I'll mass up with the order information, but that's just the consequence from chunking you have to cope with.

First, I have to thank all the people in Cognitive Unit in the University of Zurich. I won't be able to accomplish the degree without your helps. Thanks Klaus for his supervise, sorry I missed a lot of deadlines and the paper works. Thanks Corinne for translating the abstract into German and helped me move my stuff from office to office and apartment to apartment. Alexandra and Laura for helping me knowing what's on the menu. Anke for organizing all the organization works. You saved me from many nervous moments when I have to renew my resident permit every year. Michael for reviewing my abstract. My writing sucks, I know.

Second, I thank to all my family. You didn't ask me to bring a lot of stuffs from Switzerland to Taiwan, so I can keep my luggage light and small. Thanks to my parents for constantly asking when I'll finish my PhD. I won't say I finished it just to keep you quite, because I know you will find some new topic to ask about. I did it because so we can change to another subject. My girlfriend for accepting my proposal, so my parents can skip one topic. My brother for taking care of "my" cat, so the cat can stay on its course to be a pro league of legends player. My cat for being … a cat.

Third I want to thank to all my friends, in real life or online. Your names are many, and I’m not sure typing Chinese is a good idea, so I’ll just leave your name unsonged. Special shout out to Bap for calling napoc primary. Good job, captain.

Last but not least, I feel wrong for not thanking my model. Thanks for all the leg work you had done. Though I’d want you to cooperate more often, but that’s just your nature, I presume.

Curriculum Vitae

Phone: +41 44 63 57505
Cell: +41 79 26 44635
E-mail: h.lin@psychologie.uzh.ch

BIN 4.B.10. Binzmühlestrasse 14/22
CH-8050 Zurich, Switzerland

PERSONAL INFORMATION

Date of Birth: 17, July, 1983
Place of Birth: Kaohsiung City
Sex: Male
Marital Status: Unmarried
Religion: None

CURRENT

PhD Student

Cognitive Unit, Department of Psychology, University of Zurich.
Date Oct, 2011-.
Advisor: Klaus Oberauer and Björn Rasch.

EDUCATION

MS

National Chengchi University, Psychology
Date Sep, 2006- July, 2011
Concentrations: Short-term memory, Mathematical modeling.
Thesis: From Recall to Recognition: An Extension of Serial-Order in a Box Model.
Advisor: Nai-Shing Yen and Lee-Xieng Yang

PUBLICATIONS

Journal Publications

Rerko, L., Oberauer, K., & Lin, H.-Y. (2014). Spatial transposition gradients in visual working memory. *Quarterly Journal of Experimental Psychology*, *67*(1), 3–15.

Yen, N. S. *, Tsai, J. L., Chen, P. L., Lin, H. Y., & Chen , A. L. P. (2011). Effects of typographic variables on eye movement measures in reading Chinese from the screen. *Behavior & Information Technology.*

Presentations

Lin, H. Y. & Oberauer, K. (2013, Aug). Modeling visual-working memory: An interference-based model. *Talk held at 18th Meeting of the European Society for Cognitive Psychology, Budapest, Hungary.*

Lin, H. Y. * & Yang, L. X. (2010, Nov). SOB Model: Between Recognition and Recall. *Paper presented at the 49th annual meeting of the Taiwanese Psychological Association, Chiayi County, Taiwan.*

Poster

Lin, H. Y., & Oberauer, K. (2014, Aug). Recognition in Visual working memory: From Interference's Point of View. *Poster presented at the 7th European Working Memory Symposium, Edinburgh, UK.*

Lin, H. Y., & Oberauer, K. (2012, Sep). The serial-recall-based recognition: An application of Serial-Order in a Box model. *Poster presented at the 6th European Working Memory Symposium*

(EWOMS), Fribourg, Switzerland.

Lin, H. Y., Yen, N. S.* Chen, P. L. & Yu, T. Y. (2008, May). *Reading cues influence the eye movement pattern in reading Chinese.* Poster presented at the APS 20th annual convention of Association for Psychological Science, Chicago, Illinois, U. S. A. (NSC 96-2627-E-004-002-).

TEACHING

Supervising Master Thesis

Fischer, A. (2015) The Role of Categorization in Short-Term Memory.

Symposium

2013HS Modelling cognitive processes with Matlab
2014HS Modelling cognitive processes with Matlab
2015HS Modelling cognitive processes with Matlab

PROFESSIONAL TRAINING

Summer School

European Society for Cognitive Psychology Summer School in Computational and Mathematical Modeling of Cognition, Mallnitz, 2010.

AWARDS

Award of Merit, 2009

The award of merit for graduate students in National Chengchi University.

Award of Merit, 2008

The award of merit for graduate students in National Chengchi University.

RESEARCH SKILL

Matlab/Python program writing, including experiment program, data analyzing, and computational modeling.
Eye-tracker usage.
ERP equipment usage and experiment conducting.